



**Calhoun: The NPS Institutional Archive**  
**DSpace Repository**

---

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

---

1981-09

# An operational lanchester-type model of small-unit amphibious operations.

Park, Soon Dae

Monterey, California. Naval Postgraduate School

---

<http://hdl.handle.net/10945/18817>

---

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

*Downloaded from NPS Archive: Calhoun*



<http://www.nps.edu/library>

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

**Dudley Knox Library / Naval Postgraduate School**  
**411 Dyer Road / 1 University Circle**  
**Monterey, California USA 93943**

AN OPERATIONAL LANCHESTER-TYPE MODEL OF  
SMALL-UNIT AMPHIBIOUS OPERATIONS.

Soon Dae Park



# NAVAL POSTGRADUATE SCHOOL

## Monterey, California



# THESIS

AN OPERATIONAL LANCHESTER-TYPE MODEL OF  
SMALL-UNIT AMPHIBIOUS OPERATIONS

by

Soon Dae Park

September 1981

Thesis Advisor:

J. G. Taylor

Approved for public release; distribution unlimited.

T200656



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Operational Lanchester-Type Model of Small-Unit Amphibious Operations		5. TYPE OF REPORT & PERIOD COVERED Master's thesis; September 1981
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Soon Dae Park		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE September 1981
		13. NUMBER OF PAGES 100
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)  Lanchester-type combat model  Amphibious operation		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  This thesis presents an operational Lanchester-type model of small-unit amphibious operations. This relatively simple model has been developed to demonstrate the basics of model building to the beginning student interested in amphibious warfare. The model is a time sequenced, deterministic, force-on-force combat model that is implemented on a digital computer. A brief discussion of considerations for modeling amphibious operations is given. The		



## #20 - ABSTRACT - (CONTINUED)

details of the model are presented for a specific amphibious-warfare scenario. Additionally, a computer terrain-contour-line plotting program is provided to assist the combat modeler to fit a parameterized-terrain to real terrain.





Approved for public release; distribution unlimited.

An Operational Lanchester-Type Model of  
Small-Unit Amphibious Operations

by

Soon Dae Park  
Lieutenant Colonel, Republic of Korea Marine Corps  
B.S., Republic of Korean Naval Academy, 1965

Submitted in partial fulfillment of the  
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL

September 1981



## ABSTRACT

This thesis presents an operational Lanchester-type model of small-unit amphibious operations. This relatively simple model has been developed to demonstrate the basics of model building to the beginning student interested in amphibious warfare. The model is a time sequenced, deterministic, force-on-force combat model that is implemented on a digital computer. A brief discussion of considerations for modeling amphibious operations is given. The details of the model are presented for a specific amphibious-warfare scenario. Additionally, a computer terrain-contour-line plotting program is provided to assist the combat modeler to fit a parameterized-terrain to real terrain.



## TABLE OF CONTENTS

I.	INTRODUCTION -----	7
II.	GENERAL CONSIDERATIONS FOR MODELING AMPHIBIOUS OPERATIONS -----	9
	A. CHARACTERISTICS OF AMPHIBIOUS OPERATIONS ---	9
	B. MODELING APPROACH -----	11
III.	THE MODEL -----	15
	A. GENERAL -----	15
	1. The Scenario -----	15
	2. General Description of the Model -----	16
	B. THE AMPHIBIOUS ASSAULT PHASE -----	16
	1. General -----	16
	2. Attrition Process -----	21
	3. Fire Allocation -----	25
	4. Suppression -----	28
	5. The Termination of the Assault Phase ---	30
	C. THE INITIATION OF THE GROUND ATTACK -----	30
	D. THE GROUND ATTACK PHASE -----	31
	1. Movement Process -----	31
	2. Detection and Fire Allocation -----	33
	3. Attrition -----	36
	4. Termination of Ground Attack -----	38
	E. THE PARAMETRIC TERRAIN -----	39
IV.	FINAL REMARKS -----	42
APPENDIX A:	SAMPLE EXPECTED OUTPUT -----	44
APPENDIX B:	LISTING OF SAMPLE INPUT -----	47



APPENDIX C: DEFINITION OF VARIABLE IN COMPUTER PROGRAM -----	50
APPENDIX D: PROGRAM LISTING -----	56
APPENDIX E: PLOTTING PROGRAM FOR TERRAIN CONTOUR LINE -----	96
LIST OF REFERENCES -----	99
INITIAL DISTRIBUTION LIST -----	100





## I. INTRODUCTION

During the past several decades, combat models have been widely used to support military decisions. As the art of combat modeling becomes more advanced, combat modelers are continuously building more and more complicated models. To the beginning modeler, the ability to understand how those models operate is difficult, if not impossible. It is the purpose of this thesis to develop a simple amphibious-operations model that will demonstrate the basics of model building to the beginning student interested in amphibious warfare. In the broadest sense, an amphibious operation is a combined-arms operation which includes all forms of combat--land, air, sea. This thesis will limit itself to the small-unit amphibious operation.

This project started with two basic models: one was developed as the auxiliary model for the evaluation of design and employment alternatives for the LVA (Landing Vehicle Assault) in the thesis of David Larkin Chadwick (September 1978). The other is the Smoler-Mills model which was developed in the thesis of Josef Smoler (September 1979) and enriched in the thesis of Glen Mills (September 1980).

In Chapter II, general considerations for modeling amphibious operations are briefly discussed. Then, a small-scale computer-based Lanchester-type amphibious-operations model is presented, including analytical details of the algorithms used



to represent each of the combat processes considered. Although the model was developed for a specific scenario, it is sufficiently general in design so that it can be adapted to other small-scale amphibious-operations scenarios with only relatively few modifications.



## II. GENERAL CONSIDERATIONS FOR MODELING AMPHIBIOUS OPERATIONS

### A. CHARACTERISTICS OF AMPHIBIOUS OPERATIONS

In order to model an amphibious operation, it is first of all necessary to understand what is going on in a real amphibious operation. Only after one knows the details of what is happening in such a complex combat operation, can one begin to sift out the cluttering details and make valid simplifying assumptions to come up with a tractable model.

One of the key characteristics that serves to distinguish amphibious operations from other types of military operations is that a complete military force must be transferred ashore in an orderly manner under the constant pressure of actual or potential attack from hostile forces. Because the over-all amphibious assault requires precise and timely execution, the various component operations must be carried out in a planned sequence (especially early in the assault) according to a strict schedule. This sequence and these schedules, however, must be sufficiently flexible to permit rapid changes in line with unexpected development afloat and ashore.

The notable success of amphibious operations during and after World War II is testimony to the fact that, with proper planning and organization, this dual problem can be solved. In today's environment, it is well known that the modern battlefield will be dominated by highly lethal weapons. This has raised serious questions about the survivability of amphibious



forces. On the other hand, the long-range, high-speed assault potentially gives one the capability to launch assaults from far out to sea, land at times and places of one's choosing, and carry more firepower to accomplish the amphibious assault with greater safety for ships and men. Use of some type of combat model is the only way to explore such issues today. In order to build such a combat model of an amphibious operation, it is necessary to develop and consider detailed and specific information on individual tactical and support elements of the landing force, on the size, numbers, and characteristics of the equipments of these elements, and on the sequence of movement of these elements.

The amphibious operation is a combined operation, the entire spectrum of activities involved in an amphibious operation includes:

- pre-assault bombardment by ships and aircraft
- sea mine clearance
- attack on ships by enemy aircraft and cruise missile
- ship-to-shore movement
- surface assault landing
- helicopter operation
- ground combat between maneuver units
- artillery and naval gunfire support
- tactical aircraft support
- mine warfare (sea and land).

While an amphibious operation is one of the most complex of all military operations, defending against it is even more





complex: it is absolutely impossible for an enemy to defend all coastal areas at all times. The flexibility to conduct helicopter-borne vertical assault and surface-borne assault simultaneously will greatly enhance the complexity of defending against an amphibious assault.

Airborne troops and supplies were valuable during the Second War, and further developments in that direction are under way. But whether or not modern airborne tactics and techniques have supplanted (in a practical sense) seaborne assaults (such as those used from 1942 through 1945 in the Pacific and elsewhere), it should be noted that the military problem of landing forces on shores held by an enemy remains. The emphasis in the future will most likely continue to be on having the ability to project forces from the sea onto a hostile shore and to hold such a beachhead.

## B. MODELING APPROACH

All models of military operations must abstract from the real world. Since it is obvious that an engagement between modern military forces is a very complicated process, one has to abstract, aggregate, and interpolate in order to scale a combat process down to manageable size for modeling purposes. A variety of modeling approaches are available. These range from simple Lanchester-type models to highly complicated, computer assisted, high-resolution simulations in which the actions of each individual combatant are traced through a combat scenario second by second. Between these two extremes are



other approaches covering the whole spectrum of land combat, from one-on-one duels to theater-level models covering huge geographical dimensions.

There are basically four different types of combat models: war games, analytical (or mathematical), simulations and some combination of these first three types. According to Bonder [Ref. 2], war games are not a feasible mechanism for analyzing a broad spectrum of system alternatives in a responsive manner to meet a planning cycle requirement. However, they are diagnostic in the sense that they reveal problems that need to be resolved with future systems, and are a viable mechanism for training decision makers. Analytical models seek to describe the combat process mathematically. They simplify the conduct of sensitivity analysis and provide an increased ease in interpreting results, since the dynamics of the combat process are contained in readily examined equations. Analytical models of any degree of complexity usually do not yield convenient analytical solutions but require numerical approximation methods. Simulation is the most widely used technique in military system analysis. Simulation can generally produce very useful data, which are needed for further analysis, and sometimes for planning itself. However, the large amount of detail contained in most Monte Carlo simulations makes it difficult to use as the sole vehicle to single out those systems capabilities, tactics, and environmental conditions which significantly contribute to or delimit the system's effectiveness. Since, as we have seen above, no one type of combat model



is unconditionally preferred to another, it is proposed that a combat model should be selected or designed based on a specific scenario and upon analytical requirements.

In most cases, detailed models are more credible to decision makers. However, for many people such detailed models of large-scale combat operations are far too complicated to be understood, require too much input data, and (in general) are not responsive enough. When one looks at computer storage and run time requirements for even the smallest high resolution model, it is easy to see why a high resolution model of a corps or theater is presently impractical, and is likely to remain so. In order to avoid the complexity of the large-scale model and to better understand land combat there is a growing trend among analysts to combine small unit and large unit models in such a way that the output data of a high-resolution small unit combat model is used as the input data for a low-resolution large unit model. The obvious drawback of this hierarchical-modeling approach is that any errors in the small unit models will be carried through, and possibly multiplied, as the process proceeds from model to model. In the large units the emphasis has been away from simulation and towards detailed Lanchester-based models.

So far the emphasis has been adding more and more detail to the high resolution models so as to pick up as many interactions as possible. No matter how much detail is added to the small-unit simulation, it seems impossible that reality will ever be matched exactly. With this in mind, it is proposed



that a well-constructed Lanchester-based model of small unit engagements could give results that are just as valid as the results of a high-resolution simulation.





### III. THE MODEL

#### A. GENERAL

##### 1. The Scenario

The scenario considers an amphibious-landing team, consisting of reconnaissance, a light infantry unit, and landing-assault vehicles. This team is part of an Amphibious Task Force (ATF), and it disembarks from ships that are on station over the horizon from the selected landing site. The assault vehicles, after transmitting from the amphibious shipping to the designated area for the landing formation, form into conventional landing waves at a distance offshore which is greater than the effective range of the direct-fire weapon systems of the shore-defense force. During the ship-to-shore movement the defender's anti-tank guided missile and improved gunnery system respond to the landing. Naval gun-fire ships provide fire support for the assault team during the ship-to-shore movement and the initial stages of landing.

As the assault vehicles reach the beach, they (together with the assault vehicles and any weapons landed by landing vehicles) launch an attack on the enemy shore defense positions. The defenders occupying those positions fight until their losses exceed a maximum permissible amount. The attacking force, however, continues the assault irregardless of losses incurred. Once the shore assault has been completed, the landing force with tactical mobility moves inland to



carry out the tasks, while the enemy prepares to mount a counter attack.

The attacker may engage the advance force of the defender's initial counter attack force on the way to move inland. The advantage will likely go to that force which has gained the initiative (i.e., the landing force) provided it can maintain its momentum.

## 2. General Description of the Model

The model developed in this thesis is a time sequenced, deterministic, force-on-force computerized model, coded in FORTRAN. The model conducts the battle in uniform time steps of 10 seconds each. Figure 1 shows the general scheme for the sequence of events in the model. The model simulates two main phases in the amphibious operations: (I) the amphibious-assault phase, and (II) the subsequent ground attack. The framework and the logical interrelationships of these two phases will be discussed in the following subsections.

### B. THE AMPHIBIOUS ASSAULT PHASE

#### 1. General

In this phase the model considers attrition between the shore-defense force and the landing-assault force during its water-borne movement and subsequent assault to shore. The model aggregates the various actual combat organizations involved in the waterborne phase of the amphibious operation into several homogeneous combat units. Each of these units is characterized by certain relative offensive and defensive



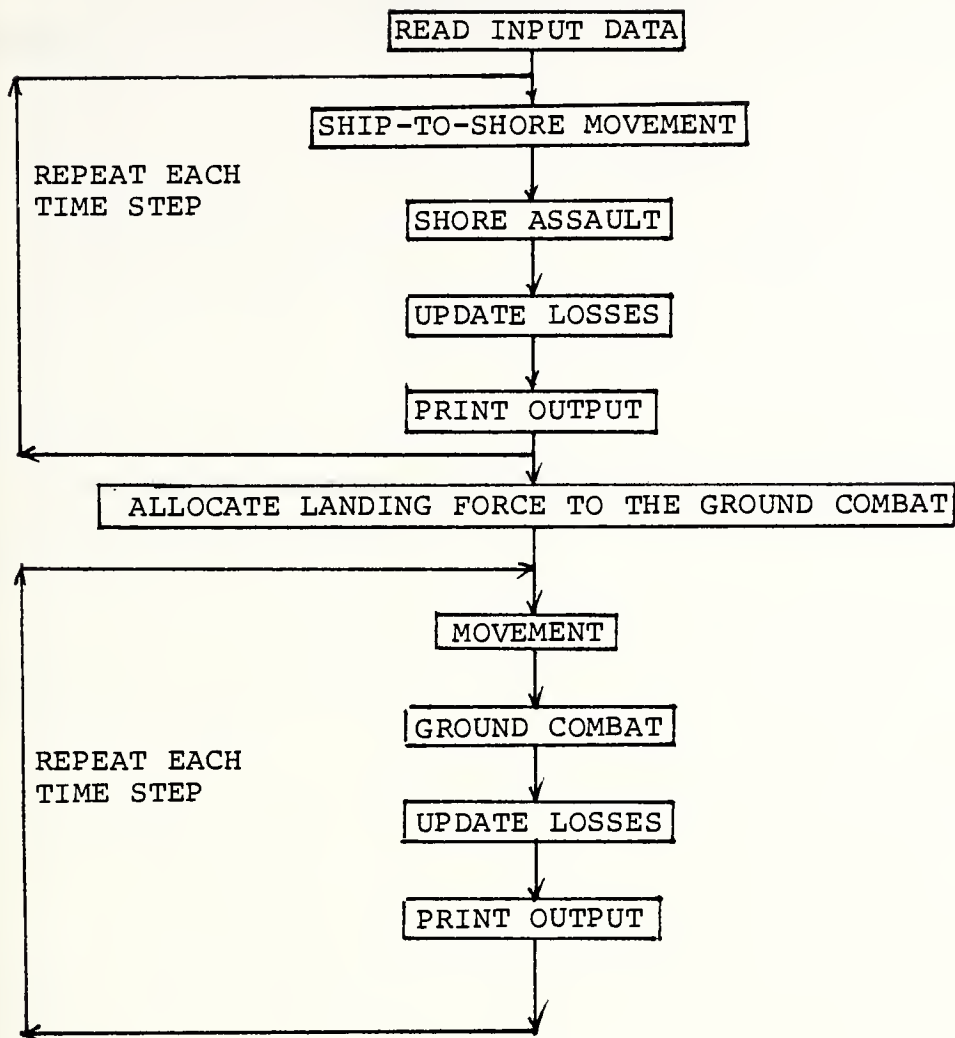


Figure 1. General Scheme for the Amphibious Operation Model



capabilities. The following table illustrates the combat organizations that were explicitly modeled. The combat strength of each unit was represented by the state variables indicated.

<u>combat organization</u>	<u>state variable</u>
Shore Defense--TANK assets	DT
Shore Defense--ATGM assets	DS
Incoming assault waves of LVA representing waves 1 through 5	WV(I), I = 1,2,3,4,5
A cumulative combat force comprised of those Marine ground units which have arrived at the beach and have debarked the LVA	TLF
Fire Support Assets of The Amphibious Task Force	ATFFS

The initial strength in each of the above units is input data to the model.

The schematic of the method of employment for the LVA in the ship-to-shore phase of an amphibious assault is shown in Figure 2. It is assumed that the conventional landing formation composed of waves of landing vehicles will be used as prescribed by current doctrine. The movement of assault vehicles to the beach is simulated using a time step approach. At each time increment the positions of vehicles are updated.

The tactical interrelationships which exist between various combat units are illustrated in Figure 3. Assuming that in such a future amphibious operation the attrition of





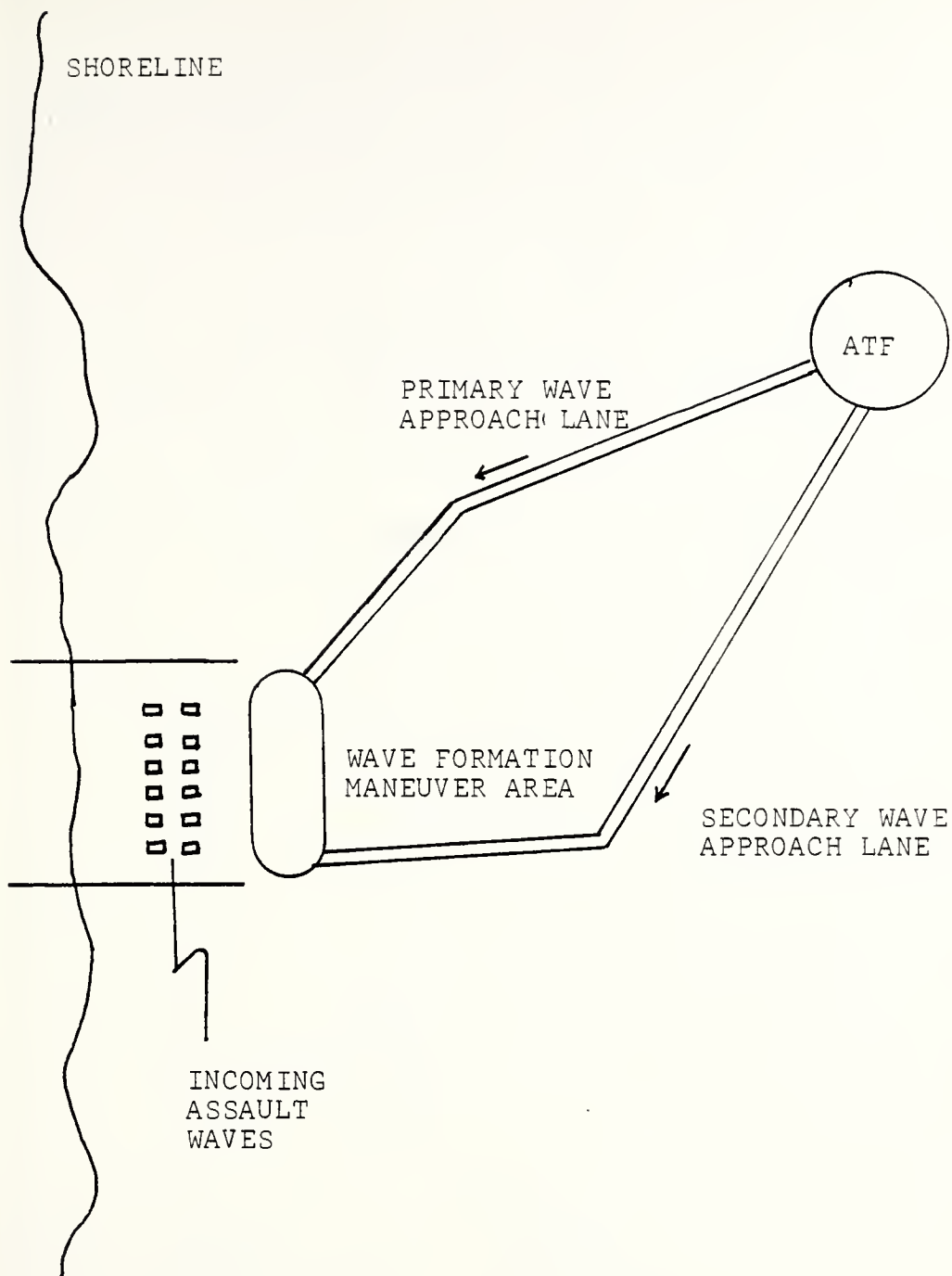


Figure 2. Concept of Ship to Shore Movement



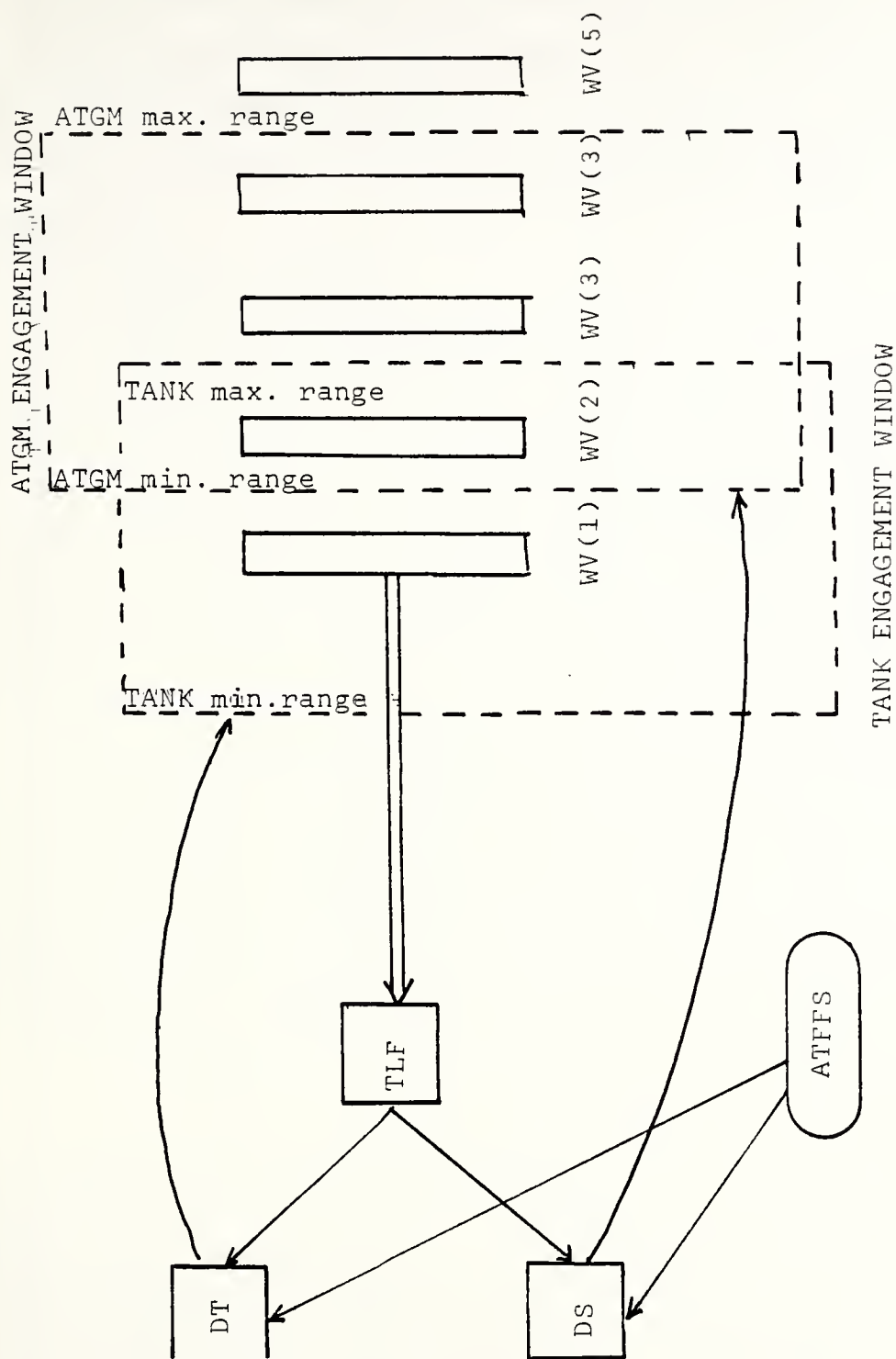


Figure 3. Schematic of Tactical Interrelationship between Combat Units during Amphibious Assault Phase



incoming landing vehicles would be dominated by the effects of shore defense direct-fire weapon systems (specifically, tank and anti-tank guided missile (ATGM) assets), the model essentially omits the effect of the defensive indirect fire capabilities.

## 2. Attrition Process

The model represents the attrition of all combatant units as a deterministic process. The primary consideration in the ship-to-shore movement of incoming waves of assault vehicle is the attrition effects upon those waves due to the direct fire weapon assets ashore. The attrition of each wave utilizes Lanchester "aimed-fire" equations with variable attrition-rate coefficients.

The classical Lanchester hypothesis for aimed-fire attrition (combat under "modern condition") is that the casualty rate of a unit is proportional to the "size" of the opposing forces. If the unit "A" is being engaged by "D", this may be expressed by the differential equation:

$$\frac{dA}{dt} = - \text{BETA}_{DA} \times D$$

The proportionality constant  $\text{BETA}_{DA}$  is called the Lanchester attrition-rate coefficient. It is assumed that this functional relationship holds for each (firing unit, target unit) pairing over a small time interval  $dt$ . The problem then is to determine numerical values for the Lanchester attrition-rate coefficients. In this model, these coefficients were expressed



as the product of the rate of fire (ROF) and the kill probability per round ( $P_k$ ). Thus,

$$BETA_{DA} = ROF_{DA} \times P(k)_{DA}$$

The rate of fire (ROF) can be expressed as the reciprocal of TBF (Time Between Firings) which can be evaluated by

$$TBF = AIM-RELOAD TIME + \frac{TARGET RANGE}{TARGET SPEED + PROJECTIVE VELOCITY}$$

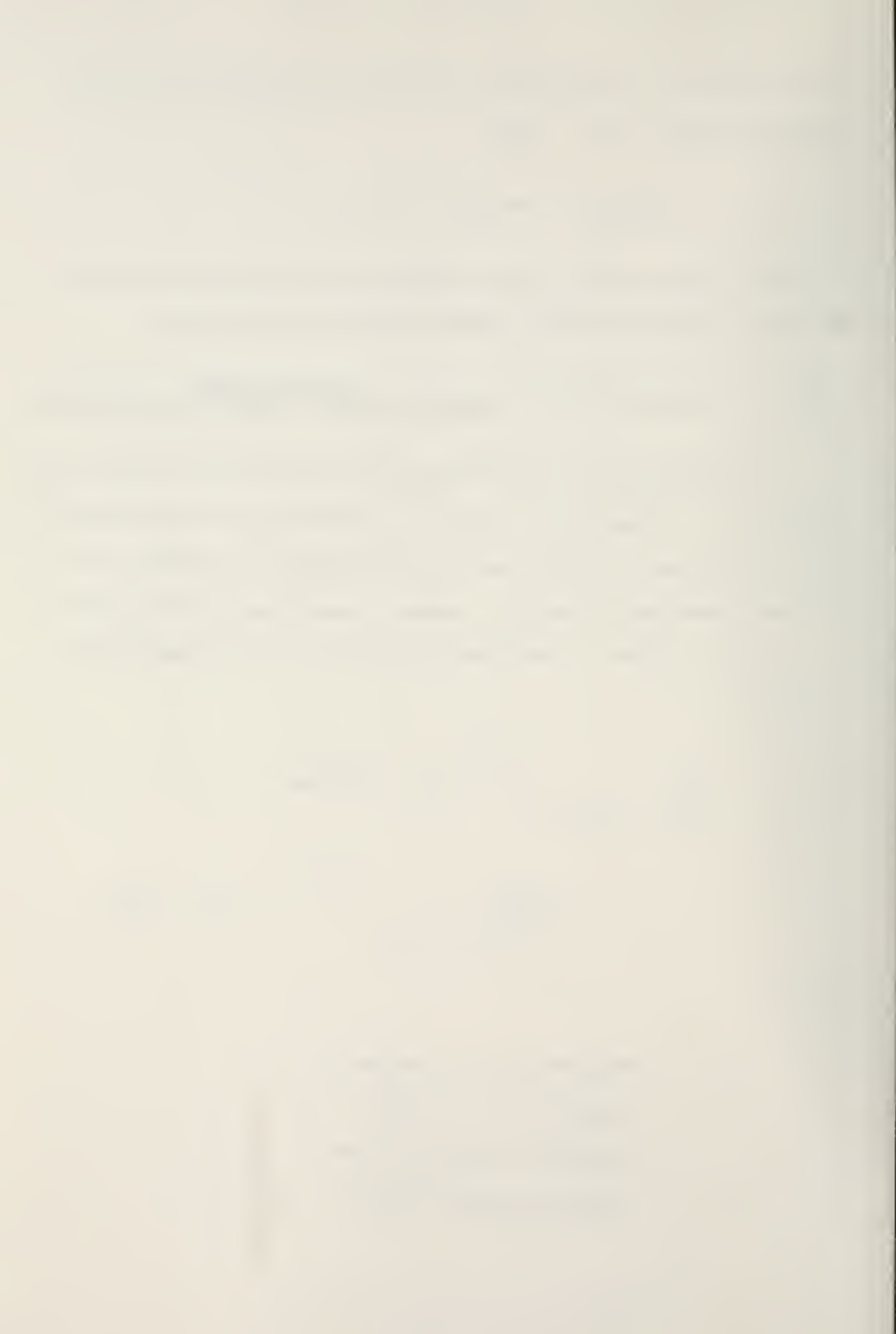
In determining the probability of a vehicle "KILL" per round, it is assumed that a hit by a large caliber projectile would constitute a "KILL" and the two defensive weapon systems addressed would exhibit normal, uncorrelated horizontal and vertical errors. Then the single shot kill probability is given by

$$P(k) = \left[ \left( \frac{1}{\sqrt{2\pi}} \right) \int_{(-a-u)/\sigma_x}^{a-u/\sigma_x} \exp\left(-\frac{x^2}{2}\right) dx \right] \cdot \left[ \left( \frac{1}{\sqrt{2\pi}} \right) \int_{(-b-v)/\sigma_y}^{(b-v)/\sigma_y} \exp\left(-\frac{y^2}{2}\right) dy \right]$$

where:

- a = semilength of a target
- b = semiwidth of a target
- u = horizontal aiming error
- v = vertical aiming error





$\sigma_x$  = round-to-round standard deviation in vertical

$\sigma_y$  = round-tround standard deviation in horizontal

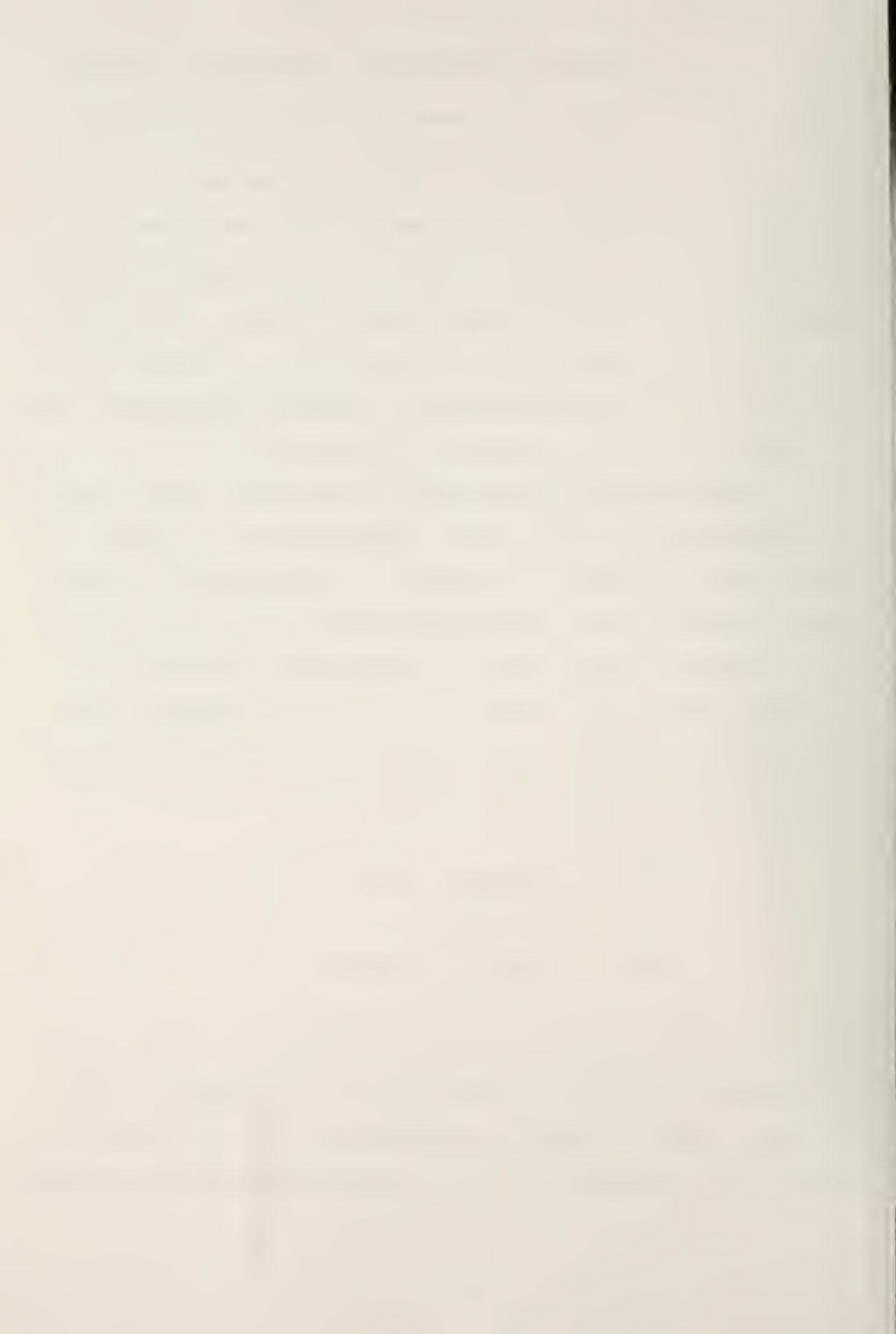
Model functions RNG, HT and SPD are called upon within the model logic to generate the range, height and speed respectively for each assault wave as time is incremented throughout the course of the amphibious assault phase. This information and typical dispersion data (both mean and standard deviation for the tank and ATGM weapons) are then incorporated into the rate of fire and hit probability calculations.

The Amphibious Task Force's fire support assets contribute significantly to the combat effectiveness of the shore defense units. Since it is assumed that the exact positions of the defensive units DT and DS emplaced on shore are unknown to the Amphibious Task Force and consequently the ATF fires into the general areas thought to contain the defensive units. The following Lanchester-type area-fire equations are applied to compute the attrition of DT and DS.

$$\frac{dDT}{dt} = -(\text{ALPHA}_{DT} \times \text{ATFFS}) \times DT$$

$$\frac{dDS}{dt} = -(\text{ALPHA}_{DS} \times \text{ATFFS}) \times DS$$

The combat effectiveness of the ATF fire support assets is to be considered relatively constant during this segment of combat time. Thus the terms in parentheses on the right hand side of these equations are to be considered an input parameter.



Once a particular defensive unit has initiated its engagement of incoming waves it is considered that their fire "gives away" their exact locations. At this point it is assumed that the ATF fires will engage that defensive unit through the use of aimed-fire and the loss rate will be in accordance with the Lanchester hypothesis for aimed fire. That is,

$$\frac{dDT}{dt} = -\text{BETA}_{DT} \times \text{ATFFS}$$

$$\frac{dDT}{dt} = -\text{BETA}_{DS} \times \text{ATFFS}$$

Again, the parameters on the right-hand sides of both these equations are provided as input.

The casualty rates applied against the DT and DS by the Total Landed Force (TLF) are determined by means of the Lanchester aimed-fire attrition rate coefficients by the equations

$$\frac{dDT}{dt} = -\text{WBETA}_{\text{TLF-DT}} \times \text{TLF}_{DT}$$

$$\frac{dDS}{dt} = -\text{WBETA}_{\text{TLF-DS}} \times \text{TLF}_{DS}$$

The computation of these WBETA coefficients is not performed within the model utilizing the detailed rate of fire and P(k) arguments described previously but is required as input. Although the defensive losses are considered significant,



a high level of complexity for computing these coefficients has not been incorporated into the model at this time.

Figure 4 describes the schematic of the attrition process of the amphibious assault phase in the model. The attrition during each time step was computed using the Euler integration method to approximate Lanchester's force-on-force attrition differential equations.

### 3. Fire Allocation

Each weapon category was assigned an engagement window as illustrated in Figure 3. Only those LVA located within these range windows could be fired upon by the shore defenders. A defensive weapon only engages the two closest incoming waves if more than two waves of LVA are at any time located within the weapon's engagement window. If only one wave of LVA is present in a weapon's engagement window, defensive fires of that particular weapon type will be distributed uniformly against the surviving LVA in that wave.

If two waves of LVA are both contained within the engagement window, defensive fires of that particular weapon type will be distributed according to a tactical allocation submodel. A weighting factor (DEFWT) is utilized in establishing the proportion of the total weapon strength to be allocated against the surviving LVAs in each of the two waves. As an example, if  $DEFWT(1) = 2$  and  $DEFWT(2) = 1$ , then each surviving LVA in the closer of the two incoming waves would be allocated twice as much fire as surviving LVA in the seaward



# DIRECT FIRE DT/DS AGAINST FOR EACH INCOMING WAVE I

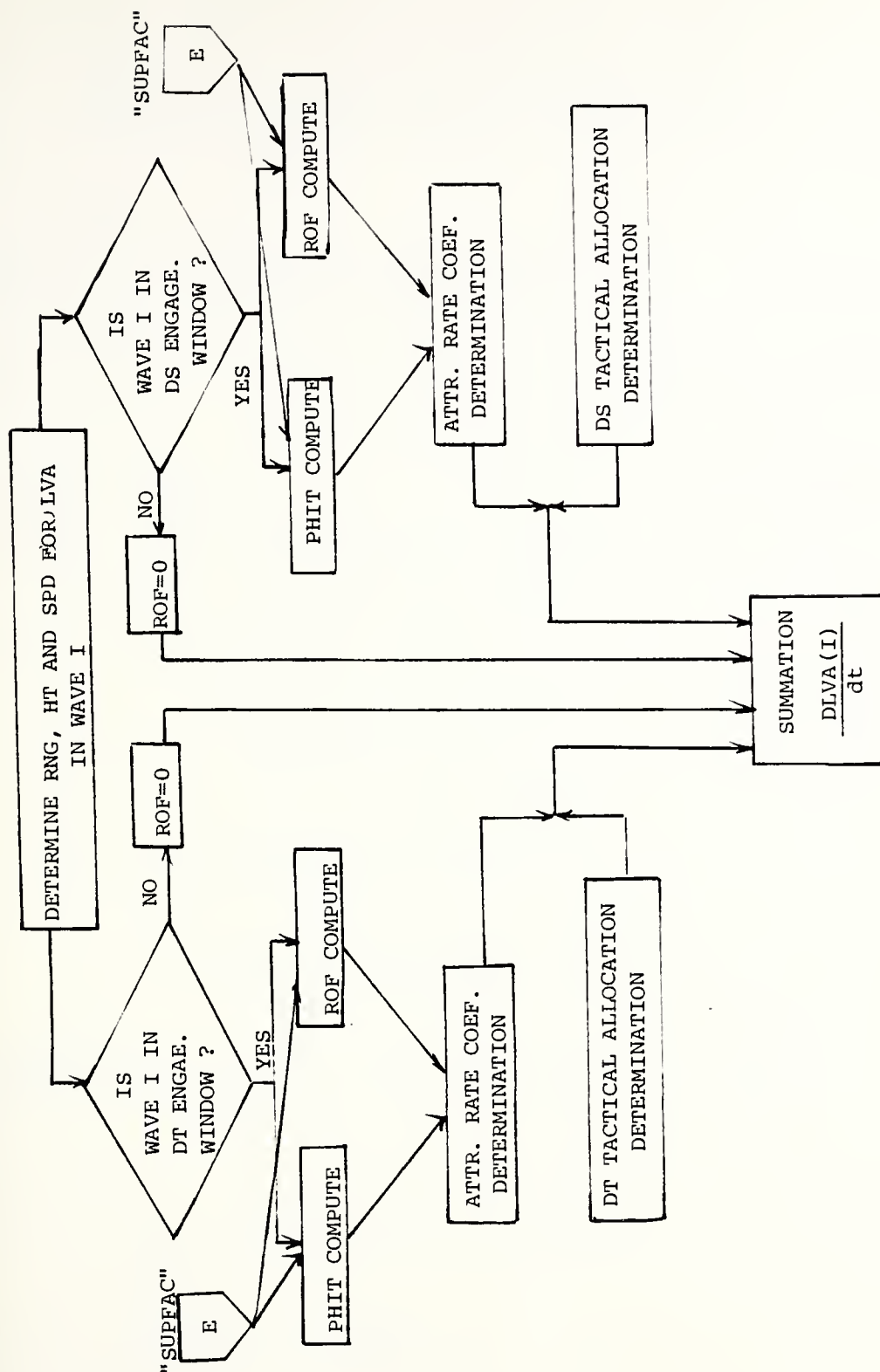


Figure 4. Schematic of The Attrition Process for The Assault Phase





# ATTRITION FOR THE SHORE DEFENSE FORCE

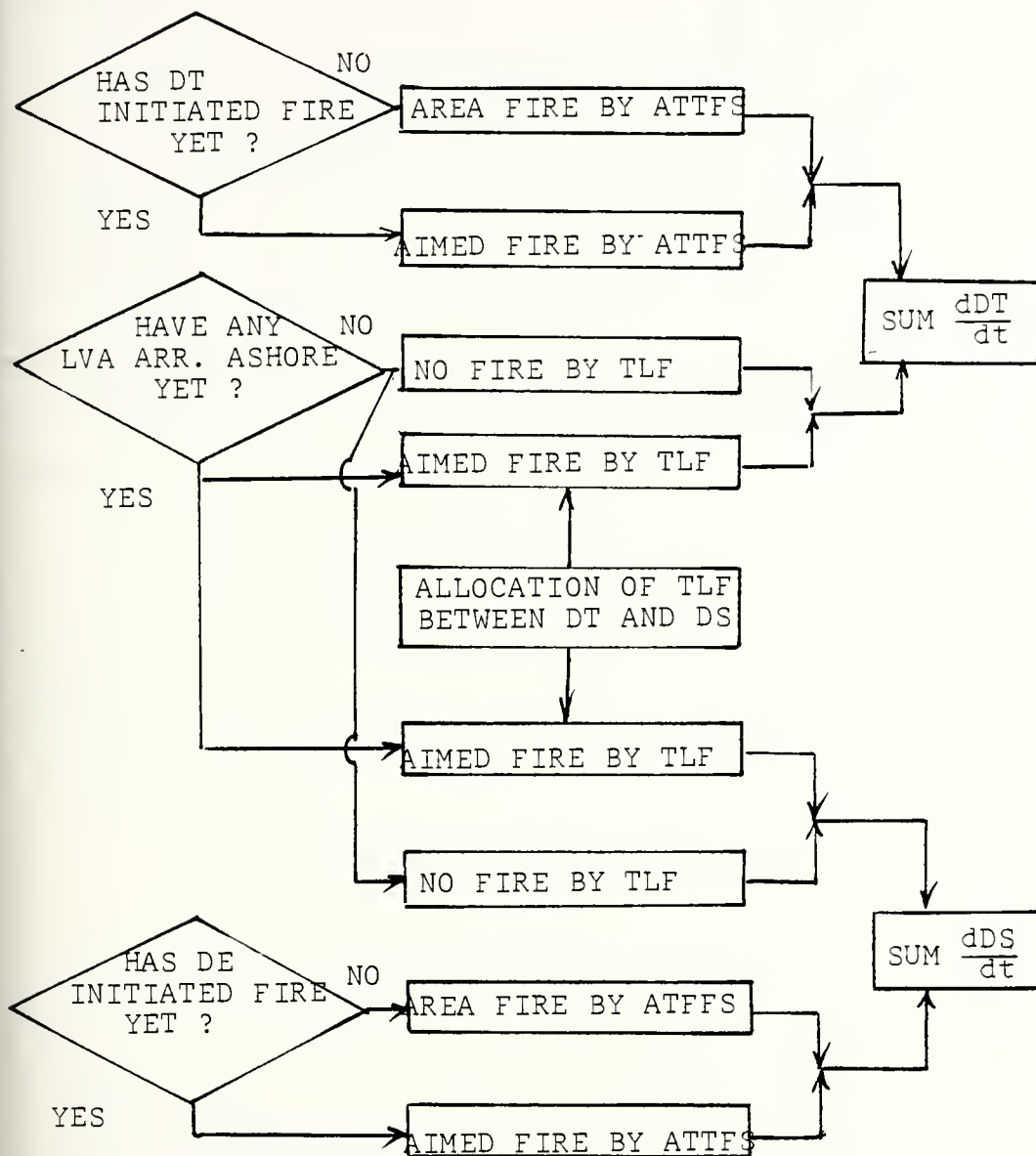


Figure 4.(cont)



wave. For the purpose of these examples, if waves 3 and 4 were both located within the tank engagement window, then the proportion of DT's fire allocated to surviving LVA in wave 3 would be

$$\frac{\text{DEFWT}(1) \times \text{WV}(3)}{\text{DEFWT}(1) \times \text{WV}(3) + \text{DEFWT}(2) \times \text{WV}(4)} \times \text{DT}$$

where WV(3) is the state variable for the current number of survivors in wave 3.

As each assault wave arrives at the beach, the total surviving strength of that wave is transferred to the variable TLF (Total Landed Force). TLF represents a ground combat force equal to that transported by the number of LVA survivors having arrived ashore. Once established, TLF engages the two shore defensive units allocating its fires between the two defensive weapon categories in the same proportion as the number of surviving tanks and ATGM's, that is

$$\text{TLF}_{\text{DS}} = \frac{\text{DS}}{\text{DT} + \text{DS}} \times \text{TLF}$$

$$\text{TLF}_{\text{DT}} = \frac{\text{DT}}{\text{DT} + \text{DS}} \times \text{TLF}$$

#### 4. Suppression

The suppression effects of incoming fire upon each of the defensive units was considered a significant factor with respect to its effect on the survivability of the incoming assault waves of LVA. Generally, the effect of suppression



fire will hinder an individual from observing and firing at the enemy.

It was assumed that suppression would degrade unit effectiveness by increasing the aim-reload time (ARTM) and round-to-round error standard deviation for each weapon system. Hypothesizing that attrition rate is the dominating variable, and therefore, a good indicator of the suppression level, ARTM and such round-to-round errors were assumed to be functions of the force's attrition rate. This is an area, however, requiring further study. Analytically,

$$ARTM_{sup} = ARTM_{nonsup}(1 + GAMMA \times DA)$$

$$ERROR_{sup} = ERROR_{nonsup}(1 + DELTA \times DA)$$

where:

DA = attrition rate for defensive unit due to the effect of AFTTS and TLF

GAMMA = parameter representing relationship between DA and ARTM

DELTA = parameter representing relationship between DA and error standard deviation

This increase of ARTM and round-to-round error (expressed as a standard deviation) decreases the kill probability ( $P_k$ ) for both defensive weapon categories. Parameter estimation would appear to be the largest problem. But, since determining these parameters in the model is beyond the scope of this thesis, these parameters GAMMA, DELTA are provided as input.



## 5. The Termination of the Assault Phase

It is assumed that if during the course of the amphibious operation the shore defense forces suffer a cumulative loss in excess of 70% of their initial force strength, the remaining shore defense will try to withdraw, resulting in termination of the engagement.

### C. THE INITIATION OF GROUND ATTACK

In the amphibious operation, the landing-force must seize critically-important inland objectives as rapidly as possible before the defenders start to react to the landing. The decision for the initiation of ground attack should be based upon the enemy threat and desired landing-force build-up ashore, among other factors. To model this decision rule, it is assumed that once the landing has begun, the landing-force commander will base his decision about initiation of ground attack primarily on the strength of the landing-force ashore and the shore defender's strength. The criteria for the decision should meet these two conditions:

- (1) The survived landing-force strength has to be greater than the minimum force required to carry out the ground attack.
- (2) The defender's strength must fall below the minimum required to continue coordinated shore defense before breakoff and retreating.

These conditions are then checked after each time step. If all waves landed without reaching the above second conditions, it





is assumed that the next wave group will engage any leftover defenders. Thus, the decision to implement the ground attack is based on the size of the total landed force.

#### D. THE GROUND ATTACK PHASE

The attacking force which is composed of three subunits of three LVAs armed with TOW antitank missile system attacks along predetermined routes. The defending force is comprised of three subunits of three tanks in a static defense.

The battle takes place on parameterized terrain which will be discussed later. The ground-attack process contains five main subprocesses: (1) movement, (2) detection, (3) fire-allocation, (4) attrition and (5) battle termination. The general flow of the ground attack phase is shown in Figure 5.

##### 1. Movement Process

Every attack unit is advanced to the next interval along a predetermined route unless this unit is destroyed already or is in firing status. To use his own determined routes, the user is required to input the original location of each attacking subunit and from one to ten nodes he wishes each attacking subunit to move through. This information, along with vehicle speed, is used to calculate route intervals that move the attacking unit through each of the designated nodes. The straight line ground distance between the first two adjacent nodes, DIST, is calculated as

$$\text{DIST} = \sqrt{X^2 + Y^2}$$



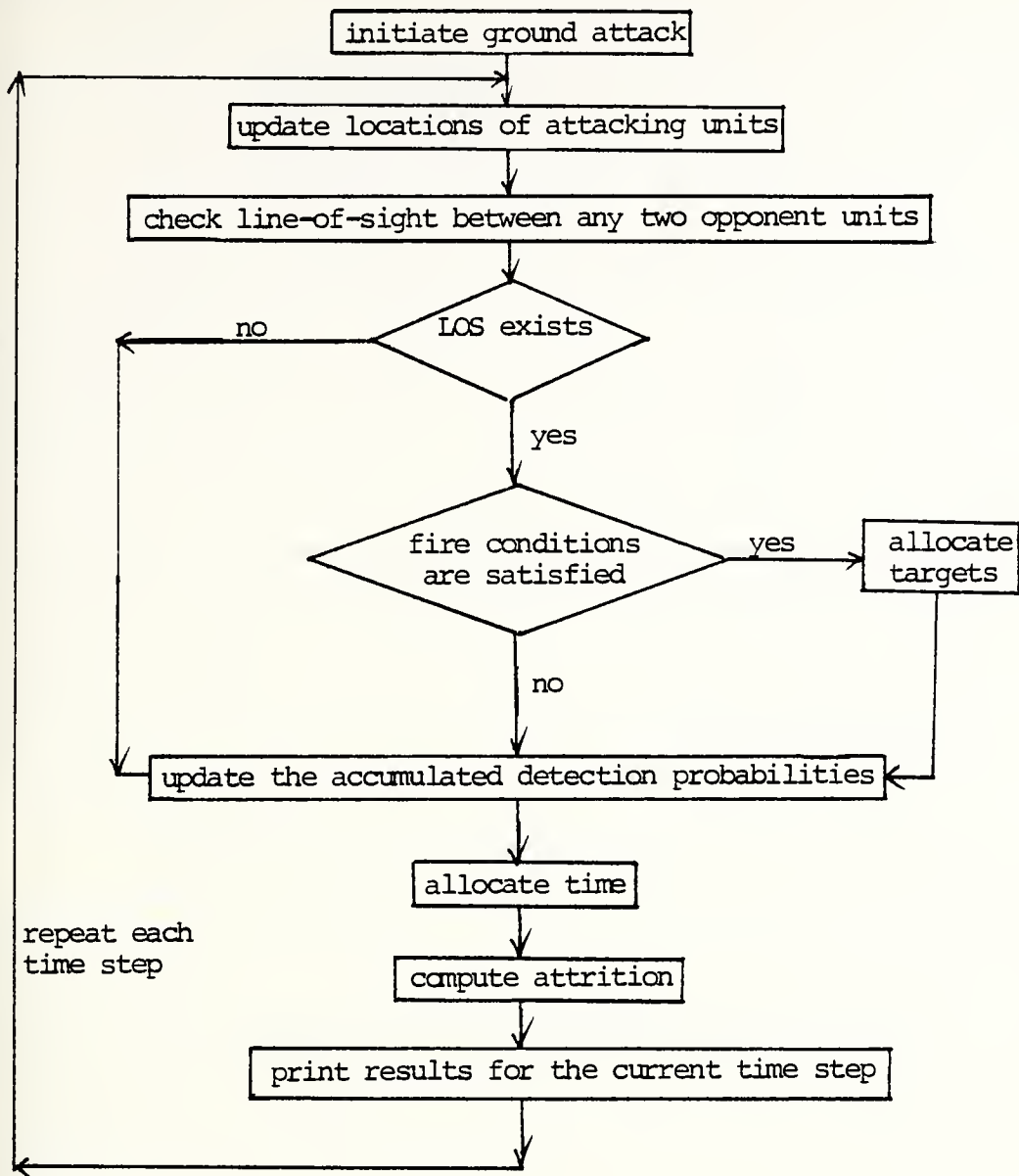


Figure 5. Flow Chart for the Ground Attack Phase



where:

X = distance between two nodes in straight west-east direction

Y = distance between two nodes in straight south to north direction

The angle between the desired direction of movement and straight west to east movement,  $\alpha$ , is then calculated. Utilizing these quantities, the distance desired to move during each time step (DST) and the distance to be moved in the X and Y direction (XLN,YLN) is computed. These distances (XLN,YLN) are then added to the coordinates of the previous interval endpoint to determine the coordinates of the next interval endpoint. This process is repeated between the next two nodes until the unit has traversed the entire route.

## 2. Detection and Fire Allocation

The detection phenomena are modeled in two ways:

(I) non-firing detection, and (II) firing-detection. A non-firing detection can occur as a result of an observer's random search within his designated sector of responsibility. The probability ( $P_k$ ) that an observer A looking at the direction which enables him to detect a target is computed by integrating the Limicon Function over limits are  $\pm 15^\circ$  from the primary direction the observer is looking. The Limicon Function,  $f(\theta)$ , is the following probability density function:

$$f(\theta) = A + B \times \cos(\theta) \quad -0 \leq \theta \leq 0$$



where:

$D = \text{assigned sector width}/2$

$A = -B \times \cos(D)$

$B = \frac{1}{2}(\sin(D) - D \times \cos(D))$

$\theta = \text{primary direction observer is looking}$

Assuming  $30^\circ$  field of view for any observer A target B might be seen only if the observer A is looking at the direction such that  $\text{ANGRT} \leq \theta \leq \text{ANGLFT}$  where:

$\text{ANGLE} = \text{the absolute value of the angle between the primary direction (IPRDIR) and the observer-target direction (OTANG)}$

$\text{ANGLFT} = \text{angle} + 15^\circ \text{ if } \text{angle} + 15 \leq D$   
 $D \text{ if } \text{angle} + 15 > D$

$\text{ANGRT} = \text{angle} - 15^\circ$

Thus

$$P_k = \text{pr}(\text{ANGRT} \leq \alpha \leq \text{ANGLFT})$$

$$= \int_{\text{ANGRT}}^{\text{ANGLFT}} f(\alpha) d\alpha$$

Given that observer A is looking at the direction, the conditional detection rate ( $\lambda_{AB}$ ) is determined by the regression curve [Ref. 11]. The probability that unit j is detected by unit i at time  $t + \Delta t$  [ $P_{ij}(t + \Delta t)$ ] is computed according to:

$$P_{ij}(t + \Delta t) = 1 - e^{-\int_t^{t+\Delta t} \lambda(t) dt}$$





$P_{ij}(t)$  can be interpreted as the average fraction of unit  $i$  that detects unit  $j$ .

The second method of detection played in the model is a so-called firing detection. This phenomena occurs when the following happens: if a firing location is within  $\pm 15^\circ$  of an observer's primary direction of observation when he is firing, he is assumed to be detected and is added to the observer's target list. In summary, the following conditions are necessary for unit  $j$  to be a target of unit  $i$ :

- (a) Line-of-sight must exist between unit  $i$  and unit  $j$ .
- (b) The range between the two units should be between maximum range and minimum range of the firer's weapon system.
- (c)  $P_{ij}(t - \Delta t) > 0$ .

The fire-allocation routine determines what fraction of each unit is allocated to fire each target in target list since it is assumed that each fire unit is not restricted to fire at one target. This fraction is determined as a function of the predetermined fire policy and  $P_{ij}(t)$ . The fire policy is as follows:

# of target	% of unit $i$ allocated to each target		
	1 <sup>st</sup> priority	2 <sup>nd</sup> priority	3 <sup>rd</sup> priority
1	100%		
2	80%	20%	
3	80%	15%	5%



The priority of a target is taken to be a function of range only. The fire allocation rule which is used in the model is documented in detail in Smoller's thesis [Ref. 9].

### 3. Attrition

The attrition process in the ground attack phase utilizes Lanchester "aimed-fire" equation used with variable attrition coefficients. The calculation of the attrition coefficients is accomplished through the use of one of two optional methods. The first option uses the following Bonder-Farrell formula to compute the reciprocal of the expected time to kill. The coefficients,  $A_{ij}$ , the rate at which one firer of unit  $i$  kills unit  $j$  targets are computed according to:

$$A_{ij} = 1/E(T_{ij})$$

where  $E(T_{ij})$  is the expected time for one firer of unit  $i$  to kill one target of unit  $j$ . The  $E(T_{ij})$  is computed using:

$$\begin{aligned} E(T_{ij}) = & t_a + t_1 - t_h + (t_h + t_f)/P(K/H) \\ & + (t_m + t_f)/P(h/m) \times ((1-P(h/h))/P(K/H) \\ & + P(h/h) - p) \end{aligned}$$

where:

$t_a$  = time to acquire a target

$t_1$  = time to fire first round following acquisition



$t_h$  = time to fire following a hit  
 $t_m$  = time to fire following a miss  
 $t_f$  = time of flight of a round  
 $P$  = probability of a first round hit  
 $P(h/h)$  = probability of a hit following a hit  
 $P(h/m)$  = probability of a hit following a miss  
 $P(k/h)$  = probability of a kill given a hit

This formula holds for the conditions that the hit probability of any round depends only on the result of the previous round and no accumulated damage is considered. It is assumed that  $P(K/H) = 1.0$  and  $P(h/m) = p(h/h) = P$ , thus reducing the equation to:

$$E(T_{ij}) = t_a + t_l + t_f + (t_m + t_f)(1-P)/P$$

The second method, called the stochastic method, interprets the attrition rate coefficient,  $A_{ij}^0$ , as a measure of the fighting ability of a unit which has a random phenomena affected by many different factors. It is assumed that the random fighting ability should be distributed between .3 and .8 with the majority of the unit being rated between .5 and .6. A "fitted" distribution to these assumed fighting levels which is devised by Mills is:

$$A_{ij}^0 = -2U^2 + 2U + .3 ; \quad U \text{ is a random Uniform } (0,1) \text{ number}$$

The  $A_{ij}^0$ 's are a realization of the random variable denoting a unit's initial fighting capability prior to the battle. Then,



during each time step, a new attrition rate coefficient for each unit is computed using the equation:

$$A_{ij} = \begin{cases} A_{ij}^0 (1 - r/r_e) & ; \text{ for } 0 \leq r \leq r_e \\ 0 & \text{ for } r_e \leq r \end{cases}$$

where:

$r_e$  = maximum effective range of a firer's weapon  
 $r$  = current range between firer and target

Utilizing one of the above formulas to calculate  $A_{ij}$ 's, the attrition during each time step was computed using the Euler-Cauchy differencing equations to approximate Lanchester's force-on-force attrition differential equations.

#### 4. Termination of Ground Attack

The ground attack is terminated when either:

- (1) One of the two opponent forces is annihilated;
- (2) A distance between each attacking subunit and each defensive subunit which is still engaging becomes "too close";
- (3) Any attacking subunit passes by the flanks of the forward most defensive subunit still in the battle.

The criteria for being "too close" is user input. This allows for flexibility of breakpoint distance for various weapon systems on the battlefield.





## E. THE PARAMETRIC TERRAIN

The terrain affects a great deal on detection, mobility, tactics, and intervisibility between weapon systems in ground combat environments. In the model, the battle is simulated on 3 x 4 Km piece of terrain represented as a part of the coastal area east of the Korean Peninsula. It is important to have a terrain representation to emulate actual terrain areas. The model uses the parametric terrain representation method which was proposed by Chris Needle [Ref. 8]. The idea of the parameterized terrain is that the elevation of any hill mass can be represented by a bivariate normal density function. Mathematically, if  $f_I(X,Y)$  is a function giving the elevation of the I's hill masses at any X,Y map coordinates on the battlefield, the overall terrain elevation at X,Y is obtained as the positive maximum over all the hill masses,

$$Z = f(X,Y) = \text{maximum } f_I(X,Y) \\ I = 1, \dots, \text{NHILLs}$$

where NHILLs is the total number of hill masses on the battlefield. Then, elevation data is used to compute the existence of line of sight between opposing forces which is a key element in detection process. The model uses the line-of-sight routine which was written by Prof. James K. Hartmann [Ref. 5].

In order to represent a piece of real terrain with parametric terrain, it is necessary to fit hill mass functions  $f_I(X,Y)$  to a contour map of the terrain to be modeled. The fitting



process can be done by comparing a computer generated contour map by varying the bivariate normal parameters to the original terrain map. The computer generated terrain map of the battle area is inclosed as Figure 6. Appendix D presents the program listing for plotting a contour map from hill mass functions. This program can be used for the user to fit a specific real terrain which he has in mind into the parameterized terrain.



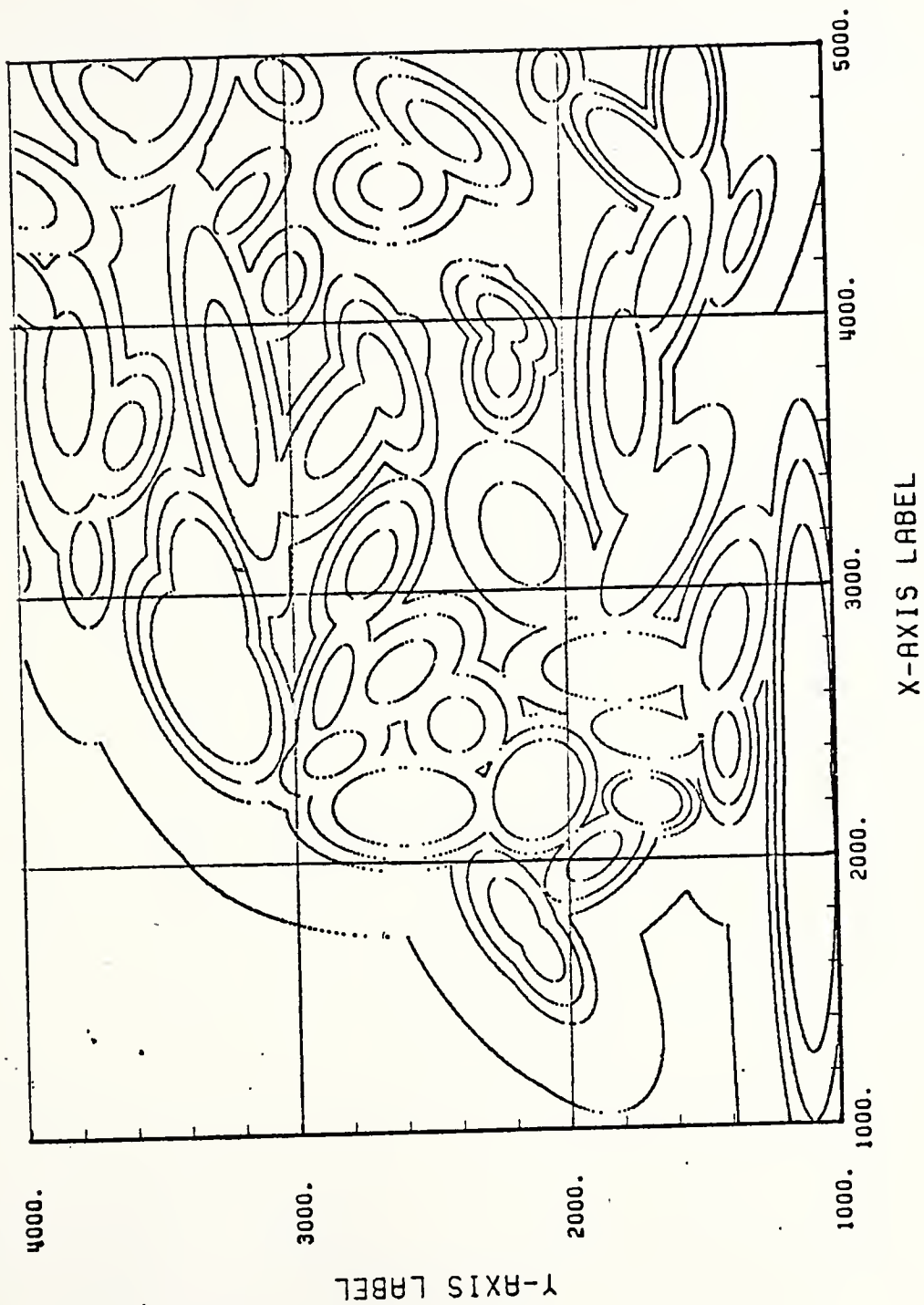


Figure 6. Model Terrain Map



#### IV. FINAL REMARKS

In order to illustrate basic modeling points for understanding and developing complex-operational amphibious-warfare models, a simplified model (specially tailored to a small-unit-amphibious-operation scenario) has been developed. Although several user options and varieties of modeling techniques have been incorporated into the model, the model should not be viewed as the final product.

There are several areas of model enhancement and enrichment that should be considered in the future. Only the aggregated amphibious task force fire support is played explicitly in the amphibious assault phase. The effect of artillery, naval gunfire and close air support during the various phases of amphibious operations should be added to the model. Ammunition consumption and resupply are considered vital to the success of any military operation, and such aspects should be also added to the model. Enemy forces should be played in greater detail. The movement of reacting enemy force and their dynamic-defensive-position selection are being considered for future model inclusion. A module which considers enemy reaction time, the terrain, and the tactical situation, and which dynamically determines which side is going to take the defensive role, as well as its defensive position, has been proposed to enhance the existing model. Inclusion of these features would create a more complicated model, adding realism but detracting from the current simple and transparent form.





The model currently simulates the ground combat on a 3 x 4 Km piece of coastal terrain representing an area of Ham-Hung, Korea. In order to simulate combat on a user's specific terrain, he needs to fit a parameterized terrain model to the actual terrain of interest. Doing so using the terrain fitting method described in the preceding chapter is an extremely tedious and time consuming process. The development of a more efficient terrain fitting technique will greatly enhance the model flexibility, and increase its utility in responding to interrogations from the real world.

The interested reader may obtain the model program deck and the sample input data deck from Prof. James G. Taylor, Naval Postgraduate School. Since the data used for the model run is hypothetical and greatly simplified, the reader is cautioned not to draw any analytical conclusions from the output.



## APPENDIX A

### SAMPLE EXPECTED OUTPUT

#### I. Amphibious Assault Summary

##### AMPHIBIOUS ASSAULT INFORMATION

##### INITIAL FORCE STRENGTH

WAVE	1	2	3	4	5
LVA	10.0	11.0	11.0	10.0	3.0

DT = 3.0                      DS = 1.0

##### LVA ENGR SPECS

SPDMAX	SPDMIN	HTMAX	HTMIN	WID
10.30	3.50	1.70	0.60	3.53

##### DEFENSIVE TACTICAL PARAMETERS

	RANGE MAX      MIN	AIM-RELOAD TIME	PROJECTILE VELOCITY
TANK	1500.0	15.00	350.00
ATGM	2000.0 200.0	30.00	350.00

##### DEFENSIVE TACTICAL ALLOCATION WEIGHTS:

WAVE 1 = 2.00    WAVE 2 = 1.00

##### DEFENSIVE FORCE ATTRITION COEFFICIENTS

	ALPHA*A	BETA*A
DT	0.00006	0.00070
DS	0.00008	0.00090

WBETA(1) = 0.00050    WBETA(2) = 0.00070

BREAKPOINT ASSUMPTION:    0.3\*(TOTAL DEP FORCE)

DEFENSIVE FORCE LEVEL FOR GROUND ATK                      0.32

##### DISPERSION DATA

RANGE	TSIGV	RANGE	TSIGH	RANGE	TMEANH
25.0	0.0	25.0	0.0	25.0	0.0
500.0	2.0	500.0	2.0	500.0	1.0
1000.0	5.0	1000.0	5.0	1000.0	5.0
2000.0	20.0	2000.0	20.0	2000.0	10.0
5000.0	25.0	5000.0	25.0	5000.0	15.0
10000.0	25.0	10000.0	25.0	10000.0	15.0



RANGE	SSIGV	RANGE	SSIGH
25.0	0.0	25.0	0.0
250.0	5.0	250.0	5.0
500.0	7.5	500.0	7.5
1000.0	14.0	1000.0	14.0
2500.0	15.5	2500.0	15.5
5000.0	17.0	5000.0	17.0
10000.0	20.0	10000.0	20.0

## II. Assault Phase Time Step Summary

TIME = 815.0 SECONDS

WAVE	FORCE LEVEL	STATUS	LOST-PCT	TSURV
1	0.7322	1	0.927	
2	1.7451	1	0.841	
3	4.5060	1	0.590	
4	9.8051	3	0.019	
5	3.0000	2	0.0	
TANK	0.0		1.000	19.79
ATGM	0.0		1.000	0.0
FINAL LVA SURVIVORS ASHORE =			19.788	
GROUND ATTACK STARTS AFTER DEFENDER BROKE CONTACT				

GROUND ATK TIME = 825.0

## III. Initial Ground Combat Summary

### INITIAL GROUND COMBAT INFORMATION

UNIT	X	Y	FORCE LEVEL
1	2000.0	1900.0	3.0
2	1900.0	2400.0	3.0
3	1500.0	2100.0	3.0
4	3800.0	2700.0	3.0
5	3800.0	2300.0	3.0
6	3600.0	1700.0	3.0

ATTRITION IS DETERMINISTIC

ROUTES DETERMINED BY USER

ATTACK VEHICLE SPEED IS 12.0

BREAKPOINT DISTANCE IS 500.0

DEFENDER WILL MOVE TO ALTERNATE POSITIONS  
ALTERNATE POSITIONS ARE:

UNIT	X	Y
4	4500.0	3800.0
5	4500.0	2700.0
6	4600.0	1800.0



# ATK KILL PROBABILITIES

RANGE	P	PHH	PHM	PKH
500	0.85	0.85	0.75	0.70
1000	0.80	0.80	0.75	0.70
1500	0.75	0.75	0.70	0.65
2000	0.60	0.65	0.60	0.55
2500	0.45	0.50	0.50	0.35
3000	0.20	0.20	0.20	0.20

# DEF. KILL PROBABILITIES

RANGE	P	PHH	PHM	PKH
500	0.60	0.70	0.65	0.85
1000	0.85	0.90	0.85	0.90
1500	0.80	0.85	0.85	0.80
2000	0.75	0.80	0.75	0.70
2500	0.60	0.70	0.65	0.65
3000	0.40	0.45	0.40	0.50

## IV. Ground Combat Time Step Summary

TIME = 1395 SECONDS

UNIT	X	Y	FORCE LEVEL	STATUS	LOST-PCT	TARGETS
1	2420.8	1984.2	0.0	2	1.000	
2	3664.1	2253.0	0.0	2	1.000	
3	4397.4	1742.2	2.9	0	0.038	
4	4500.0	3800.0	0.0	2	1.000	
5	4500.0	2700.0	2.8	0	0.055	
6	4600.0	1800.0	0.0	2	1.000	

\* DISTANCE BETWEEN FORCES IS TOO CLOSE. END OF BATTLE





# APPENDIX B

## LISTING OF SAMPLE INPUTS

### I. Amphibious Assault Input

1	1					
10.30	3.5	1.7	0.6	3.533		
10.						
1500.	2000.	200.				
15.	30.	350.	350.			
25.	500.	1000.	2000.	5000.	10000.	0.
2.	5.	20.	25.	25.		
25.	500.	1000.	2000.	5000.	10000.	0.
2.	5.	20.	25.	25.		
25.	500.	1000.	2000.	5000.	10000.	0.
1.	5.	10.	15.	15.		
25.	250.	500.	1000.	2500.	5000.	10000.
0.	5.	7.5	14.	15.5	17.	20.
25.	250.	500.	1000.	2500.	5000.	10000.
0.	5.	7.5	14.	15.5	17.	20.
2.	1.					
10.	11.	11.	10.	3.		
3.	1.					
0.00006	0.00008					
0.0007	0.0009					
0.0005	0.0007					
.32						
50.	100.					

### II. Terrain Data

46					
0.					
2000.	1100.	170.	0.1	999.9	8.0
1800.	2200.	150.	30.	350.	2.0
2000.	1900.	150.	130.	300.	2.
2400.	1400.	150.	0.1	300.	2.5
2450.	1700.	130.	80.	500.	2.2
2700.	1800.	138.	90.	500.	2.2
3200.	1650.	140.	150.	600.	3.
4300.	1300.	130.	160.	400.	3.5
3750.	1750.	150.	0.1	660.	3.6
4150.	1600.	150.	160.	550.	3.
3200.	2150.	130.	25.	500.	1.5
4600.	1700.	170.	45.	300.	2.5
4800.	1500.	170.	0.1	300.	2.5
2200.	2600.	170.	90.	350.	1.8
2400.	2850.	150.	120.	300.	1.8
3100.	2700.	150.	150.	350.	2.



2500.	2400.	150.	0.1	250.	1.0
2650.	2850.	150.	160.	400.	3.0
2700.	2600.	150.	130.	370.	1.8
3800.	2200.	150.	0.1	230.	1.5
4500.	2600.	150.	90.	280.	1.3
3600.	2800.	150.	145.	500.	2.5
2700.	3300.	190.	25.	350.	2.0
3000.	3300.	170.	15.	400.	2.5
3150.	3750.	130.	0.1	350.	2.5
3750.	3200.	150.	10.	850.	5.0
3800.	3800.	150.	0.1	650.	3.
3600.	3600.	150.	160.	320.	3.0
4150.	3950.	170.	30.	220.	2.2
1650.	2100.	150.	30.	300.	2.0
2250.	2100.	180.	150.	220.	1.2
4000.	2200.	150.	45.	280.	2.
3900.	2200.	150.	0.1	300.	3.5
0	0	0	0	1	7
0	33	39	53	62	0
0	0	0	0	0	6
0	6	14	9	12	0
101					
1	2	3	30	4	43
6	32	33	7	11	31
8	9	10	11	33	43
8	42	2	14	30	23
16	17	18	19	20	3
2	31	11	16	20	22
46	20	21	22	12	34
42	45	46	14	23	15
26	14	25	26	27	28
35	44	26	27	29	35
40					

### III. Ground Combat Input

```

1 28943
03 03
0000.0 2500.0 0500. 4000.0
3.0 3.0 3.0
1 2
2000.0 1900.0
1900.0 2400.0
1500.0 2100.0
01
5000.0 2500.0
01
4900.0 2150.0
02
2200.0 1700.0
4800.0 1750.0
3800.0 2700.0 3.0 190 120

```



3800.0	2300.0	3.0	190	120
3600.0	1700.0	3.0	180	120
0	0500.0	4		
4500.0	3800.0			
4500.0	2700.0			
4600.0	1800.0			
0.85	0.85	0.75	0.70	
0.80	0.75	0.70	0.65	
0.75	0.75	0.70	0.65	
0.60	0.65	0.60	0.55	
0.45	0.50	0.50	0.35	
0.20	0.20	0.20	0.20	
0.60	0.70	0.65	0.85	
0.85	0.90	0.84	0.90	
0.80	0.85	0.85	0.80	
0.75	0.80	0.75	0.70	
0.60	0.70	0.65	0.65	
0.40	0.45	0.40	0.50	



## APPENDIX C

### DEFINITION OF VARIABLES IN COMPUTER PROGRAM

#### 1. The Amphibious Assault Phase

CDSURV(I) = Current strength of defensive force I

I = 1 TANK

I = 2 ATGM

CSURV(I) = Current strength of assault wave I

DA(I) = Attrition rate for def. unit I due to the effects of ATFFS/TLF

DS1 = That portion of the DS unit assigned to engaging the closer of two multiple waves in the ATGM engagement window

DS2 = That portion of the Ds unit assigned to engaging the farther of two multiple waves in the ATGM engagement window

DT1 = That portion of the Dt unit assigned to engaging the closer of two multiple waves in the TANK engagement window

DT2 = That portion of the DT unit assigned to engaging the farther of two multiple waves in the TANK engagement window

DT1PH = Hit probability of rounds fired by DT1 against wave TENG(1)

DT1ROF = Rate of fire utilized by DT1 against wave TENG(1)

DINIT = Initial strength of def. force I

IL(I) = When equal to 1 indicates the wave landed shore

IWPN = Weapon code: TANK = 1, ATGM = 2

IWSTAT(I) = Current status of wave I

0 - not engaging

1 - landed

2 - under fire by ATGM

3 - under fire by TANK

4 - under fire both ATGM and TANK





GALF = Denote whether the LF build-up is sufficient  
         for the ground attack  
         0 - not sufficient  
         1 - sufficient

GATK = Denote whether the LF initiated the ground attack  
         0 - not started yet  
         1 - started already

GATM = Time when the ground attack started

RD = Distance offshore at which waves initiate  
      transition

RKSURV(I) = Concatenation of CSURV and CDSURV

SA(I) = Attrition rate for wave I due to ATGM

SENG(I) = The wave number of the closer of two waves in  
          the ATGM engagement window

SRNG(I) = Firing range to wave SENG(I)

SSIGH = The std dev error in the horizontal for ATGM

SSIGV = The std dev error in the vertical for ATGM

SWTS(I) = The proportion of the total DS strength to be  
          allowed to engaging SENG(I)

TA(I) = Attrition rate for wave I due to TANK

TBW = The interarrival time between waves arriving at  
      the beach

TMEANH = The bias error in the horizontal for TANK

TMEANV = The bias error in the vertical for TANK

TENG(I) = The wave number of the closer of two waves in  
          the tank engagement window

TRNG(I) = the firing range to wave TENG(I)

TSIGH = The std dev error in the horizontal for TANK

TSIGV = The std dev error in the vertical for TANK

TSURV = Total number of surviving LVA at the current time

TWTS(I) = The proportion of the total DT strength to be  
          allowed to engaging TENG(I)

WVINT(I) = Initial strength of wave I



## 2. The Ground Attack Phase

ALPHA(I)	=	Initial attrition-rate coefficient for stochastic attrition module
APOA(I,J)	=	The average proportion of the $j^{\text{th}}$ attacker of unit i allocated to fire on unit i
AVSP	=	Average speed of moving attacking units
BREAK	=	Breakpoint distance between attacking units and defenders
DISMAX	=	Maximum distance allowed between attacking units before the leading unit is delayed
DIST	=	The straight-line distance between two movement nodes inputed by the user
DST	=	The distance in meters to be moved each time step by attacking units
FL(I)	=	Force level of unit i
FO(I)	=	Force level of unit i
FO(I)	=	Initial force level of unit i
IALT	=	Denotes whether the user desires alternate defensive positions or not 0 - yes 1 - no
IC	=	Counts number of time units a defender has been moving
IDIR(I,J)	=	Direction of $j^{\text{th}}$ interval in $i^{\text{th}}$ route
II(I)	=	Interval index for unit i
IMOVE	=	Number of time units a defender is allowed for moving to an alternate position
IPRDIR(I)	=	Primary direction of movement for unit i
IRTE	=	Denotes whether user wants to input routes or not 0 - program determined routes 1 - user determined routes
IS	=	Random number seed used for stochastic attrition
ISECWD(I)	=	Width of search sector for unit i



ISPD = Input variable to denote user's desired speed for attackers movement  
1 - 9 mph  
2 - 12 mph  
3 - 15 mph  
4 - 18 mph

ITEM = Input variable denoting number of time steps allowed for defender's move

ITIME = Current time, in seconds, of battle

ITRIT = Input variable denoting whether attrition will be stochastic or deterministic  
0 - stochastic  
1 - deterministic

IUSTAT(I) = Current status of unit i  
0 - unit alive, not firing  
1 - unit alive and firing  
2 - unit killed  
3 - unit moving

LOA(I,J) = The number of the  $j^{\text{th}}$  attacker of unit i

LOST(I,J) = Denotes whether line-of-sight exists between unit i and j or not

LOT(I,J) = The number of the  $j^{\text{th}}$  target of unit i

MVTDIR(I) = Movement direction of unit i

N(I) = Number of nodes inputed by user for route i

NA(I) = Number of attackers of unit i

NBU = Number of defense units

NF(I) = Number of time units i is allowed to fire at the same location

NLOSC(I,J) = Number of continuous time steps that line-of-sight does not exist between unit i and unit j

NOI(I) = Number of intervals in the  $i^{\text{th}}$  route

NRU = Number of attack units

NT(I) = Number of targets of unit i

OFL(I) = Force level of unit i during previous time step



$P(I,J)$	= Probability of 1 <sup>st</sup> round hit by unit i in range band j
$PHH(I,J)$	= Probability of a hit following a hit by unit i in range band j
$PHM(I,J)$	= Probability of a hit following a miss by unit i in range band j
$PKH(I,J)$	= Probability of a kill given a hit by unit i in range band j
PM	= The proportion of time a moving unit is searching for targets
$POA(I,J)$	= The proportion of the j <sup>th</sup> attacker of unit i allocated to fire on unit i
$POL(I)$	= Percent of unit i lost during the current time step
$PTT(I)$	= Proportion of surviving firepower allocated to the i <sup>th</sup> target if there are j targets available
RANGE	= Current minimum distance between attackers and defenders
$Q(I,J)$	= Probability that unit j is not detected by unit i at current time
RF	= Detection rate reduction factor for a firing unit (in comparison with non-firing unit)
RMINTK	= Minimum effective range for defending weapon system
RMINTW	= Minimum effective range for attacking weapon system
RMXTK	= Maximum effective range for defending weapon system
RMXTW	= Maximum effective range for attacking weapon system
$ROT(I,J)$	= The range of the j <sup>th</sup> target of unit i
SIZETK	= Size of attacking vehicle
SIZETW	= Size of defending vehicle
$TA(K)$	= Time to acquire a target for k <sup>th</sup> weapon system type (k = 1,2)
$TF1(K)$	= Time of flight to 1000m for k <sup>th</sup> weapon system type (k = 1,2)





TF2(K) = Time of flight to 2000m for  $k^{\text{th}}$  weapon system type ( $k = 1,2$ )  
 TF3(K) = Time of flight to 3000m for  $k^{\text{th}}$  weapon system type ( $k = 1,2$ )  
 TH(K) = Time to fire a round following a hit for weapon system type  $k$  ( $k = 1,2$ )  
 TI(K) = Time to fire first round after target has been acquired for weapon system type  $k$  ( $k = 1,2$ )  
 TM(K) = Time to fire a round following a miss for weapon system type  $k$  ( $k = 1,2$ )  
 TNKFR = Firing rate for attacking weapon system  
 TOWFR = Firing rate for defending weapon system  
 TPOL(I) = Total percentage of lost since battle began for unit  $i$   
 VISFR(I,J) = The fraction of unit  $i$  seen by unit  $j$   
 VISFRA = Fraction of unit A as seen by unit B  
 VISFRB = Fraction of unit B as seen by unit A  
 X(I),Y(I) = Coordinates of unit  $i$   
 XA(I),YA(I) = Coordinates of alternate position for defender  $i$   
 XIC(I,J) = Coordinates of the  $j^{\text{th}}$  interval endpoint of the route for unit  $i$   
 YIC(I,J)  
 XL,YL = Distance added to previous interval endpoint for vehicle to move DST during a time step  
 XLOC(I,J) = Coordinates of the  $j^{\text{th}}$  node input by the user for the route of unit  $i$   
 YLOC(I,J)



# APPENDIX D

## PROGRAM LISTING

```

C  AMPHIBIOUS ASSAULT PHASE
COMMON /AMPH/ IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DIN,IT(2),GAINL,IMSTAT(5)
COMMON /ENGR/ SPDMAX,HTMIN,TTSTAA,TB,TFF
COMMON /DISPER/TSIGV(6,2),TSGH(6,2),TMEANH(6,2),
1SSIGV(7,2),SSIGH(7,2)
COMMON /DEF/TENGMX,SENGMX,TARTM,SARTM,TVEL,
1SVEL,DEFWTS(2)
COMMON /SUPEFT/GAMMA,DELTA
COMMON /IOUT/ISURV,IATTR
C  GROUND ATTACK PHASE
COMMON /GRP1/ IPRDIR(6),ISECWD(6),MVTDIR(6),X(6),Y(6),SPD(6)
COMMON /GRP2/ TA(2),T1(2),TH(2),TF1(2),TF2(2),TF3(2),
1P(2,6),PHH(2,6),PHM(2,6),PKH(2,6),TF(2)
COMMON /GRP3/ NBU,NRU,FL(6),FO(6),NOI(3),XIC(3,200),YIC(3,200),
1IDIR(3,200),AVSP,ISPD
COMMON /GRP4/ VISFRA,VISFRB,SIZETK,
1IUSTAT(6),I1(6),LOST(6,6),D1SMAX,
1SIZETW,NT(6),INF(6),SRF,D1SMAX,
INLOSC(6,6),VISFR(6,6),RMINTK,RMXTW,OP,TOWFR,TNKFR,
1PTTT(3,3),RF,POA(6,6),APOA(6,6),LOA(6,6),OFL(6),POL(6)
COMMON /GRP5/ TPOL(6),OLDQ(6,6),Q(6,6)
COMMON /HILLS/ LOT(6,6),ROT(6,6)
COMMON /HILLS/ XC(100),YC(100),PEAK(100),ANGH(100),SPRD(100)
COMMON /HILLS/ ECC(100),PXX(100),PXY(100),BASE
COMMON /HILLS/ NHILLS
COMMON /COVER/ CXC(150),CPEAK(150),CPXX(150),CPYY(150)
COMMON /COVER/ CPXY(150),NCVELS
COMMON /COUNTR/ KH,KHW,KV,KN,KGRS,KELL,KINT
COMMON /GRID/ LST(5,4),NHL(5,4),LISTH(150),KHREP(150),KTREP
COMMON /GRID/ LSTC(5,4),NC(5,4),LISTC(400),KCREP(150)
COMMON /GRP6/ ALPHA(6)
COMMON /GRP7/ XA(6),YA(6),IMOVE(6)

GATM=0.
GATK=0.
CALL DATIN
CALL SETUP
CALL SEA(GATM,GATK)
IF(GATK.NE.0.) GO TO 106
WRITE(6,105)
105 FORMAT(IX,'TOTAL LANDED LF STRENGTH IS NOT SUFFICIENT FOR GATK')

```



```

106 STOP
107 IF(GATK-2.0) 10,20,30
108 WRITE(6,107)
109 FORMAT(IX,'GROUND ATTACK STARTS WHILE SHORE COMBAT IS GOING ON')
110 GO TO 110
111 GO TO 110
112 WRITE(6,108)
113 FORMAT(IX,'GROUND ATTACK STARTS AFTER DEFENDER BROKE CONTACT')
114 GO TO 110
115 GO TO 110
116 WRITE(6,109)
117 FORMAT(IX,'GROUND ATTACK STARTS AFTER ALL WAVES LANDED')
118 WRITE(6,111) GATM
119 FORMAT(IX,'GROUND ATK TIME=',F6.1)
120 CALL GROUND(GATM)
121 STOP
122 END

```

C

```

SUBROUTINE SEA(GATM,GATK)
COMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DINIT(2),GAINL,IWSTAT(5)
COMMON /ENGR/ SPDMAX,SPDMIN,HTMAX,HTMIN,TTS,TAA,TB,TFF
CALL OUTPUT
IRD=500
ITBW=120
RD=1.0*IRD
TBW=1.0*ITBW
TINT=0.0

```

C

```

COMPUTATION OF FIRST WAVE TIME PARAMETERS
TA-TIME FIRST WAVE INITIATES TRANSITION
TB-TIME FIRST WAVE COMPLETES TRANSITION
TFF-TIME FIRST WAVE REACHES THE BEACH

```

C

```

TAA=(5000.-RD)/SPDMAX
TB=TAA+TTS
TFF=TB+(RD-(0.5*(SPDMAX-SPDMIN)*TTS)-150.)/SPDMIN
DEL=10.
WRITE(6,55) RD,TBW
55 FORMAT(IX,'ITERATION INITIATED...RD=',F10.3,IX,'TBW=
1.,F10.3)
CALL RKINT(DEL,TINT,N,GATM,GATK)
RETURN
END
SUBROUTINE RKINT(H,TI,N,GATM,GATK)

```

C

```

SUBROUTINE RKINT PROVIDES THE INTERFACE BETWEEN
THE EULER NUMERICAL INTEGRATION ROUTINE(RKLDQ)
AND THE SUBROUTINE ATTR WHICH DETERMINES EACH
UNIT'S STATUS AS TIME PROGRESSES THROUGH THE

```

C



```

C      AMPHIBIOUS OPERATION
C      CCOMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1      ITBW,DIN IT(2),GAINL,IWSTAT(5)
C      CCOMMON /IOUT/ISURV,IATTR
C      DIMENSION CSURV(5),CDSURV(2),TA(5),SA(5),DA(2)
C      DIMENSION RKSURV(7),RKATTR(7),TATTR(200,12),TIME(200)

C      IMAX - MAXIMUM ALLOWABLE NUMBER OF TIME INTERVALS
C      ITE - A SWITCH VARIABLES SET TO 1 WHEN THE DEF.TANK
C      UNIT INITIATES ITS FIRE
C      ISE - A SWITCH VARIABLES SET TO 1 WHEN THE DEF.ATGM
C      UNIT INITIATES ITS FIRE
C      T - CURRENT TIME
C      IT - CURRENT TIME PERIOD

C      GALF=0.
C      IMAX=199
C      ITE=0
C      ISE=0
C      TSURV=0.
C      TIME(I)=0.
C      T=TI
C      DO 10 I=1,5
C      CSURV(I)=WVINT(I)
C      TSURV=TSURV+CSURV(I)
C      IL(I)=0
C      IWSTAT(I)=0
10      CONTINUE
C      DO 15 I=1,2
C      CDSURV(I)=DINIT(I)
15      CONTINUE
C      DO 20 J=1,12
C      TATTR(I,J)=0.
20      IT=1
C      DO 25 I=1,5
C      RKSURV(I)=CSURV(I)
25      RKSURV(6)=CSURV(1)
C      RKSURV(7)=CSURV(2)
C      DO 30 I=1,7
C      RKATTR(I)=0.
30      NT=0
1000      CALL ATTR(T,CSURV,CDSURV,TA,SA,DA,GALF,GATK,GATM)
C      IF(IL(1).EQ.99) GO TO 1200
C      DO 40 I=1,5
C      RKSURV(I)=CSURV(I)
C      RKATTR(I)={TA(I)+SA(I)}*(-1.0)
40      DO 45 I=1,2

```

```

GRA00900
GRA00910
GRA00920
GRA00930
GRA00940
GRA00950
GRA00960
GRA00970
GRA00980
GRA00990
GRA01000
GRA01010
GRA01020
GRA01030
GRA01040
GRA01050
GRA01060
GRA01070
GRA01080
GRA01090
GRA01100
GRA01110
GRA01120
GRA01130
GRA01140
GRA01150
GRA01160
GRA01170
GRA01180
GRA01190
GRA01200
GRA01210
GRA01220
GRA01230
GRA01240
GRA01250
GRA01260
GRA01270
GRA01280
GRA01290
GRA01300
GRA01310
GRA01320
GRA01330
GRA01340
GRA01350
GRA01360
GRA01370

```





GRA01380  
 GRA01390  
 GRA01400  
 GRA01410  
 GRA01420  
 GRA01430  
 GRA01440  
 GRA01450  
 GRA01460  
 GRA01470  
 GRA01480  
 GRA01490  
 GRA01500  
 GRA01510  
 GRA01520  
 GRA01530  
 GRA01540  
 GRA01550  
 GRA01560  
 GRA01570  
 GRA01580  
 GRA01590  
 GRA01600  
 GRA01610  
 GRA01620  
 GRA01630  
 GRA01640  
 GRA01650  
 GRA01660  
 GRA01670  
 GRA01680  
 GRA01690  
 GRA01700  
 GRA01710  
 GRA01720  
 GRA01730  
 GRA01740  
 GRA01750  
 GRA01760  
 GRA01770  
 GRA01780  
 GRA01790  
 GRA01800  
 GRA01810  
 GRA01820  
 GRA01830  
 GRA01840  
 GRA01850

```

45  RKSURV(I+5)=CDSURV(I)
    RKATTR(I+5)=-1.0*DA(I)
    S=RKLEDEQ(7,RKSURV,RKATTR,T,H,NT)
    DO 50 I=1,5
      CSURV(I)=RKSURV(I)
    CONTINUE
50  DO 55 I=1,2
      CDSURV(I)=RKSURV(I+5)
    CONTINUE
55  IF(S-1.) 1100,1000,1200
1100 WRITE(6,60)
60  FORMAT(IX,'ERRO..S.NE.1.OR.2')
    STOP
1200 CONTINUE
    IT=IT+1
    TSURV=0.
    DO 65 L=1,5
      TSURV=TSURV+CSURV(L)
65  IF(TSURV.LE.0.) TSURV=0.
    TIME(IT)=T
C  PRINT RESULT OF SHIP TO SHORE MOVEMENT AFTER EACH TIME STEP
112  FORMAT(//IX,'TIME=',F6.1,IX,'SECONDS'//)
113  WRITE(6,113)
113  FORMAT(IX,'WAVE',2X,'FORCE LEVEL',2X,'STATUS',2X,'LOST-PCT',
113  12X,'TSURV')
    DO 66 I=1,4
      PLOST=1.-CSURV(I)/WVINT(I)
      WRITE(6,114) I,CSURV(I),IWSTAT(I),PLOST
114  FORMAT(3X,I1,3X,F10.4,5X,I1,5X,F8.3)
66  CONTINUE
      PLOST=1.-CSURV(5)/WVINT(5)
      WRITE(6,115) CSURV(5),IWSTAT(5),PLOST,TSURV
115  FORMAT(3X,I5,3X,F10.4,5X,I1,5X,F8.3,2X,F5.2)
      PLOST=1.-CDSURV(1)/DINIT(1)
      WRITE(6,116) CDSURV(1),PLOST
116  FORMAT(IX,'TANK',2X,F10.4,11X,F8.3)
      PLOST=1.-CDSURV(2)/DINIT(2)
      TASURV=CDSURV(1)+CDSURV(2)
      WRITE(6,117) CDSURV(2),PLOST,TASURV
117  FORMAT(IX,'ATGM',2X,F10.4,11X,F8.3,2X,F5.2)
C
    DO 80 J=1,5
      TATTR(IT,J)=TA(J)
80  TATTR(IT,J+5)=SA(J)
    DO 85 J=1,2
      TATTR(IT,J+10)=DA(J)
85  R=RNG(T-4.*TBW)
  
```



GRA01860  
GRA01870  
GRA01880  
GRA01890  
GRA01900  
GRA01910  
GRA01920  
GRA01930  
GRA01940  
GRA01950  
GRA01960  
GRA01970  
GRA01980  
GRA01990  
GRA02000  
GRA02010  
GRA02020  
GRA02030  
GRA02040  
GRA02050  
GRA02060  
GRA02070  
GRA02080  
GRA02090  
GRA02100  
GRA02110  
GRA02120  
GRA02130  
GRA02140  
GRA02150  
GRA02160  
GRA02170  
GRA02180  
GRA02190  
GRA02200  
GRA02210  
GRA02220  
GRA02230  
GRA02240  
GRA02250  
GRA02260  
GRA02270  
GRA02280  
GRA02290  
GRA02300  
GRA02310  
GRA02320  
GRA02330

```

C  DETERMINE IF ALL WAVES LANDED AND GROUND ATK STARTED
C  IF(R.LT.75.) GO TO 2000
    IF(IT.GT.IMAX) GO TO 2000
    IF(IL(1).EQ.99) GO TO 2000
    GO TO 1000
2000 N=IT
    WRITE(6,90) TSURV
    FORMAT(IX,F10.3)
    IF(GATK.GE.1.) GO TO 2222
    IF(TSURV.LT.9.) GO TO 2222
    GATK=3.
    GATM=T
    RETURN
2222
    FUNCTION RKLDEQ(N,Y,F,X,H,NT)
    DIMENSION Y(1),F(1),Q(25)
    NT=NT+1
    GO TO (1,2,3,4),NT
    1  H1=H
       AA=H1/4.0
       DO 11 J=1,N
       11 Q(J)=0.
       X=X+AA
       GO TO 5
       2  X=X+AA
       GO TO 5
       3  X=X+AA
       GO TO 5
       4  DO 93 L=1,N
       93 Y(L)=Y(L)+AA*F(L)
       NT=0
       X=X+AA
       RKLDEQ=2.
       GO TO 6
       5  DO 90 I=1,N
       90 Y(I)=Y(I)+AA*F(I)
       6  RETURN
    END

C  SUBROUTINE ATTR(T,CSURV,DSURV,TA,SA,DA,GALF,GATK,GATM)
C  GIVEN THE CURRENT TIME AND STATE VARIABLE STRENGTHS,
C  SUBROUTINE ATTR DETERMINES THE ATTRITION RATES AND UPDATES
C  THE STATUS OF EACH UNIT WITH RESPECT TO SHORE MOVEMENT
C  AND IMPLEMENTS THIS INFORMATION INTO THE ATTRITION LOSS RATE
C  COMPUTATION.

```



```

C TA(I) - CURRENT ATTRITION LOSS RATE FOR WAVE I DUE TO TANK FIRE
C SA(I) - CURRENT ATTRITION LOSS RATE FOR WAVE I DUE TO ATGM FIRE
C DA(I) - CURRENT ATTRITION LOSS RATE FOR DEF. FORCE I DUE TO
C         ATFFS(AMPHIBIOUS TASK FORCE FIRE SUPPORT)/TLF EFFECTS
C
C      CCOMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RO,WVINT(5),WID,
C      1TBW,DIN IT(2),GAINL,IWSTAT(5)
C      CCOMMON /DEF/TENGMX,SENGMX,TARTM,SARTM,TVEL,
C      1SVEL,DEFWTS(2)
C      1INTEGER TENG(2),SENG(2)
C      1DIMENSION TRNG(2),TWTS(2),SRNG(2),DSURV(2),SWTS(2),
C      1CSURV(5),TA(5),SA(5),DA(2)
C
C      DO 10 I=1,5
C      TA(I)=0.
C      SA(I)=0.
C      10 CONTINUE
C
C      DS1=0.
C      DS2=0.
C      DT1=0.
C      DT2=0.
C      FAC=1.0
C
C      DETERMINE IF PART OF LANDING FORCE ADVANCE TO ATTACK INLAND
C      KEY TERRAIN
C
C      IF(GATK.EQ.1.0) GO TO 2929
C      IF(GALF.EQ.1.0.AND.(DSURV(1)+DSURV(2)).LE.GAINL*(DINIT(1)
C      1+DINIT(2))) GATM=1
C      IF(GALF.EQ.1.0.AND.(DSURV(1)+DSURV(2)).LE.GAINL*(DINIT(1)
C      1+DINIT(2))) GATK=1.0
C
C      DETERMINE IF DEF. BREAKPOINT HAS BEEN REACHED
C
C      2929 IF((DSURV(1)+DSURV(2)).LT.0.3*(DINIT(1)+DINIT(2)))
C      1 GO TO 20
C
C      DETERMINE ATTRITION RATE ON DEFENSIVE FORCES BY ATFFS
C      BASED UPON AREA OR AIMED FIRE STATUS
C
C      DA(1)=B(1)
C      DA(2)=B(2)
C      IF(ITE.EQ.0) DA(1)=A(1)*DSURV(1)
C      IF(ISE.EQ.0) DA(2)=A(2)*DSURV(2)
C      GO TO 30

```

```

GRA02340
GRA02350
GRA02360
GRA02370
GRA02380
GRA02390
GRA02400
GRA02410
GRA02420
GRA02430
GRA02440
GRA02450
GRA02460
GRA02470
GRA02480
GRA02490
GRA02500
GRA02510
GRA02520
GRA02530
GRA02540
GRA02550
GRA02560
GRA02570
GRA02580
GRA02590
GRA02600
GRA02610
GRA02620
GRA02630
GRA02640
GRA02650
GRA02660
GRA02670
GRA02680
GRA02690
GRA02700
GRA02710
GRA02720
GRA02730
GRA02740
GRA02750
GRA02760
GRA02770
GRA02780
GRA02790
GRA02800
GRA02810

```



```

20 DSURV(1)=0.
   DSURV(2)=0.
   DA(1)=0.
   DA(2)=0.
   IF(GATK.EQ.1.) GO TO 3939
   GAT=1
C
C DETERMINE IF DEF.BREAKPOINT HAS BEEN REACHED BEFORE SUFFICIENT
C LANDING FORCE IS BUILT UP ON THE SHORE FOR INLAND ATTACK
C
97 DO 91 I=1,5
   WVRNG=RNG(GAT-TBW*(I-1))
   IF(WVRNG.LT.75.) IL(I)=1
   IF(IL(I).EQ.1) TLF=TLF+CSURV(I)
91 CONTINUE
   GAT=GAT+10.
   IF(TLF.LT.9.0.AND.IL(5).EQ.1) RETURN
   IF(TLF.LT.9.0.AND.IL(5).NE.1) GO TO 97
   GATK=2.
   GATF=1.
   GATM=GAT
   WRITE(6,220) GATM
220 FORMAT(7,1X,'GROUND ATK INITIATES AT TIME=',F7.1)
3939 IL(1)=99
25 WRITE(6,25) T
25 FORMAT(1X,'BEAKPOINT REACHED AT TIME = ',F9.3)
   RETURN
30 CALL DTGTS(T,TENG,TRNG,TWTS,SENG,SRNG,SWTS,CSURV)
C
C DETERMINE THE CUMULATIVE NUMBER OF SURVIVING LVA'S
C THAT HAVE BEEN REACHED THE BEACH - TLF
C
   TLF=0.
   DO 40 J=1,5
   IF(IL(J).EQ.1) TLF=TLF+CSURV(J)
40 CONTINUE
C
C DETERMINE IF TLF BUILT UP IS SUFFICIENT FOR GROUND ATK
C
   IF(TLF.GE.9.) GATF=1.
   IF(GATK.EQ.1.) TLF=TLF-9.
C
C ALLOCATE THE FORCE STRENGTH OF TLF BETWEEN THE TWO
C DEFENSIVE FORCE UNITS
C
   DSUM=DSURV(1)+DSURV(2)
   TLF1=(DSURV(1)/DSUM)*TLF
   TLF2=(DSURV(2)/DSUM)*TLF

```

GRA02820  
 GRA02830  
 GRA02840  
 GRA02850  
 GRA02860  
 GRA02870  
 GRA02880  
 GRA02890  
 GRA02900  
 GRA02910  
 GRA02920  
 GRA02930  
 GRA02940  
 GRA02950  
 GRA02960  
 GRA02970  
 GRA02980  
 GRA02990  
 GRA03000  
 GRA03010  
 GRA03020  
 GRA03030  
 GRA03040  
 GRA03050  
 GRA03060  
 GRA03070  
 GRA03080  
 GRA03090  
 GRA03100  
 GRA03110  
 GRA03120  
 GRA03130  
 GRA03140  
 GRA03150  
 GRA03160  
 GRA03170  
 GRA03180  
 GRA03190  
 GRA03200  
 GRA03210  
 GRA03220  
 GRA03230  
 GRA03240  
 GRA03250  
 GRA03260  
 GRA03270  
 GRA03280  
 GRA03290





```

C ADD TO DA1 AND DA2 THE ATTRITION LOSS RATE DUE
C TO THE EFFECTS OF TLF1 AND TLF2
C
C      DA{1}=DA{1}+TLF1*WB{1}
C      DA{2}=DA{2}+TLF2*WB{2}
C      IF(DSURV{1}.LE.0.0) DA{1}=0.0
C      IF(DSURV{2}.LE.0.0) DA{2}=0.0
C
C DETERMINE IF THERE EXISTS AN INCOMING WAVE IN THE
C TANK ENGAGEMENT WINDOW I.E. TENG(1).NE.0
C      IF(TENG(1).EQ.0.) GO TO 100
C      ITE=1
C
C DETERMINE THE TIME SINCE WAVE TENG(1) CROSSED THE
C 5000. METER OFFSHORE MARK -T1
C      T1=T-TBW*(TENG(1)-1)
C      DT1=TWTS(1)*DSURV(1)
C      FAC=1.
C
C DETERMINE THE SUPPRESSION EFFECT TO BE IMPOSED
C ON THE DT UNIT BASED ON THE ATTRITION LOSS RATE
C CURRENTLY IN EFFECT
C      SUPFAC=DA(1)
C
C      CALL RATE(TRNG(1),SPD(T1),1,SUPFAC,DT1ROF)
C      CALL PHIT(TRNG(1),WID,HT(T1),1,SUPFAC,DT1PH)
C
C DETERMINE THE ATTRITION LOSS RATE FOR WAVE TENG(1)
C DUE TO DT1 FIRES
C      TA(TENG(1))=DT1PH*DT1ROF*DT1
C
C DETERMINE IF THERE IS A SECOND INCOMING WAVE THAT
C IS IN THE TANK ENGAGEMENT WINDOW, IF THERE IS THE
C ATTRITION RATE COMPUTATIONS ARE SIMILAR IN FORM
C TC THOSE PREVIOUSLY PERFORMED FOR THE CLOSER WAVE
C
C      IF(TENG(2).EQ.0) GO TO 100
C      T2=T-TBW*(TENG(2)-1)
C      DT2=TWTS(2)*DSURV(1)
C      CALL RATE(TRNG(2),SPD(T2),1,SUPFAC,DT2ROF)
C      CALL PHIT(TRNG(2),WID,HT(T2),1,SUPFAC,DT2PH)
C      TA(TENG(2))=DT2PH*DT2ROF*DT2
C
C

```

GRA03300  
 GRA03310  
 GRA03320  
 GRA03330  
 GRA03340  
 GRA03350  
 GRA03360  
 GRA03370  
 GRA03380  
 GRA03390  
 GRA03400  
 GRA03410  
 GRA03420  
 GRA03430  
 GRA03440  
 GRA03450  
 GRA03460  
 GRA03470  
 GRA03480  
 GRA03490  
 GRA03500  
 GRA03510  
 GRA03520  
 GRA03530  
 GRA03540  
 GRA03550  
 GRA03560  
 GRA03570  
 GRA03580  
 GRA03590  
 GRA03600  
 GRA03610  
 GRA03620  
 GRA03630  
 GRA03640  
 GRA03650  
 GRA03660  
 GRA03670  
 GRA03680  
 GRA03690  
 GRA03700  
 GRA03710  
 GRA03720  
 GRA03730  
 GRA03740  
 GRA03750  
 GRA03760  
 GRA03770



```

C C DETERMINE IF THERE EXISTS AN INCOMING WAVE IN THE ATGM
C C ENGAGEMENT WINDOW, IF THERE IS, DETERMINE THE ATTRITION
C C EFFECTS AGAINST THAT WAVE DUE TO ATGM THE ATTRITION
C C RATE COMPUTATION ARE SIMILAR IN FORM TO THOSE FOR THE
C C EFFECTS DUE TO THE TANK FIRE.
C
100 IF(SENG(1).EQ.0) GO TO 200
   ISE=1
   SI=T-TBW*(SENG(1)-1)
   DS1=SWTS(1)*DSURV(2)
   SUPFAC=DA(2)
   CALL RATE(SRNG(1),SPD(S1),2,SUPFAC,DS1ROF)
   CALL PHIT(SRNG(1),WID,HT(T1),2,SUPFAC,DS1PH)
   SA(SENG(1))=DS1PH*DS1ROF*DS1
   IF(SENG(2).EQ.0) GO TO 200
   S2=T-TBW*(SENG(2)-1)
   DS2=SWTS(2)*DSURV(2)
   CALL RATE(SRNG(2),SPD(S2),1,SUPFAC,DS2ROF)
   CALL PHIT(SRNG(2),WID,HT(S2),2,SUPFAC,DS2PH)
   SA(SENG(2))=DS2PH*DS2ROF*DS2
200 RETURN
C
C SUBROUTINE DTGTS(T,TENG,TRNG,TWTS,SENG,SRNG,SWTS,CSURV)
C
C GIVEN THE CURRENT TIME AND LVA WAVE SURVIVOR POPULATIONS,
C SUBROUTINE DTGTS DETERMINES THE WAVE NUMBERS THAT ARE
C TO BE ENGAGED BY THE DT AND DS DEFENSIVE UNITS BASED
C ON THE ENGAGEMENT WINDOW CRITERIA
C
COMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DINIT(2),GAINL,IWSTAT(5)
COMMON /DEF/TENGMX,SENGMX,SENGMN,TARTM,SARTM,TVEL,
1SVEL,DEFWTS(2)
INTEGER TENG(2),SENG(2)
DIMENSION TRNG(2),SRNG(2),TWTS(2),SWTS(2),CSURV(5),DEMO(5)
DO 10 I=1,2
  TENG(I)=0
  TWTS(I)=0
  TRNG(I)=0
  SRNG(I)=0
  SENG(I)=0
  SWTS(I)=0
10 CONTINUE
  JT=0
  JS=0
  JSUM=0.
  SSUM=0.

```

GRA03780  
 GRA03790  
 GRA03800  
 GRA03810  
 GRA03820  
 GRA03830  
 GRA03840  
 GRA03850  
 GRA03860  
 GRA03870  
 GRA03880  
 GRA03890  
 GRA03900  
 GRA03910  
 GRA03920  
 GRA03930  
 GRA03940  
 GRA03950  
 GRA03960  
 GRA03970  
 GRA03980  
 GRA03990  
 GRA04000  
 GRA04010  
 GRA04020  
 GRA04030  
 GRA04040  
 GRA04050  
 GRA04060  
 GRA04070  
 GRA04080  
 GRA04090  
 GRA04100  
 GRA04110  
 GRA04120  
 GRA04130  
 GRA04140  
 GRA04150  
 GRA04160  
 GRA04170  
 GRA04180  
 GRA04190  
 GRA04200  
 GRA04210  
 GRA04220  
 GRA04230  
 GRA04240  
 GRA04250



```

DO 100 I=1,5
  WVRNG=RNG(I-TBW*(I-1))
  IF(WVRNG.LT.75.) IL(I)=1
  IF(WVRNG.LT.75.) IWSTAT(I)=1
C
C IF THE FIRING RANGE TO A WAVE IS LESS THAN 75 METERS,
C THE WAVE IS CONSIDERED TO HAVE REACHED A COVERED AND
C CONCEALED POSITION ON THE BEACH
C
  IF((WVRNG.GT.TENGMX).OR.(CSURV(I).LT.0.05).OR.
    1(WVRNG.LT.75.).OR.(JT.GE.2)) GO TO 50
    JT=JT+1
    TENG(JT)=I
    TWTS(JT)=DEFWTS(JT)*CSURV(I)
    TSUM=TSUM+TWTS(JT)
    TRNG(JT)=WVRNG
  50 IF((WVRNG.GT.SENGMX).OR.(CSURV(I).LT.0.05).CR.
    1(WVRNG.LT.SENGMN).OR.(JS.GE.2)) GO TO 100
    JS=JS+1
    SENG(JS)=I
    SRNG(JS)=WVRNG
    SWTS(JS)=DEFWTS(JS)*CSURV(I)
    SSUM=SSUM+SWTS(JS)
  100 CONTINUE
C
C DETERMINE WAVE STATUS
C
  DO 20 I=1,2
    DO 25 J=1,5
      IF(IWSTAT(I,J).NE.1.AND.SENG(I).EQ.J) IWSTAT(J)=2
    25 CONTINUE
  20 CONTINUE
  DO 30 I=1,2
    DO 35 J=1,5
      IF(IWSTAT(I,J).EQ.1) GO TO 35
      IF(IWSTAT(I,J).EQ.2.AND.TENG(I).EQ.J) IWSTAT(J)=4
      IF(IWSTAT(I,J).NE.2.AND.TENG(I).EQ.J) IWSTAT(J)=3
    35 CONTINUE
  30 CONTINUE
C
  IF(TENG(1).EQ.0) GO TO 500
  DO 200 I=1,2
    TWTS(I)=TWTS(I)/TSUM
  200 CONTINUE
  500 IF(SENG(1).EQ.0) RETURN
  DO 600 I=1,2
    SWTS(I)=SWTS(I)/SSUM
  600 CONTINUE

```

GRA04260  
 GRA04270  
 GRA04280  
 GRA04290  
 GRA04300  
 GRA04310  
 GRA04320  
 GRA04330  
 GRA04340  
 GRA04350  
 GRA04360  
 GRA04370  
 GRA04380  
 GRA04390  
 GRA04400  
 GRA04410  
 GRA04420  
 GRA04430  
 GRA04440  
 GRA04450  
 GRA04460  
 GRA04470  
 GRA04480  
 GRA04490  
 GRA04500  
 GRA04510  
 GRA04520  
 GRA04530  
 GRA04540  
 GRA04550  
 GRA04560  
 GRA04570  
 GRA04580  
 GRA04590  
 GRA04600  
 GRA04610  
 GRA04620  
 GRA04630  
 GRA04640  
 GRA04650  
 GRA04660  
 GRA04670  
 GRA04680  
 GRA04690  
 GRA04700  
 GRA04710  
 GRA04720  
 GRA04730



RETURN  
END

C

```

SUBROUTINE DATAIN
COMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DINIT(2),GAINL,IWSTAT(5)
COMMON /ENGR/SPDMMAX,HTMAX,HTMIN,TTS,TAA,TB,TFF
COMMON /DISPER/TSIGV(6,2),TSIGH(6,2),TMEANH(6,2),
1SSIGV(7,2),SSIGH(7,2)
COMMON /DEF/TENGMX,SENGMX,SENGMN,TARTM,SARTM,TVEL,
1SVEL,DEFWTS(2)
COMMON /SUPEFT/GAMMA,DELTA
COMMON /IOUT/ISURV,IATTR
READ(5,50) ISURV,IATTR
READ(5,100) SPDMMAX,SPDMMIN,HTMAX,HTMIN,WID
READ(5,100) TTS
READ(5,100) TENGMX,SENGMX,SENGMN
READ(5,100) TARTM,SARTM,TVEL,SVEL
READ(5,100) ((TSIGV(I,J),I=1,6),J=1,2)
READ(5,100) ((TSIGH(I,J),I=1,6),J=1,2)
READ(5,100) ((TMEANH(I,J),I=1,6),J=1,2)
READ(5,100) ((SSIGV(I,J),I=1,7),J=1,2)
READ(5,100) ((SSIGH(I,J),I=1,7),J=1,2)
READ(5,100) (DEFWTS(I),I=1,2)
READ(5,103) (WVINT(I),I=1,5)
READ(5,100) (DINIT(I),I=1,2)
READ(5,101) (A(I),I=1,2)
READ(5,101) (B(I),I=1,2)
READ(5,101) (WB(I),I=1,2)
READ(5,110) GAINL
READ(5,101) GAMMA,DELTA
FORMAT(F5.2)
110 FORMAT(2I5)
150 FORMAT(7F10.3)
100 FORMAT(2F10.5)
101 FORMAT(5F10.5)
103 RETURN
END

```

110  
150  
100  
101  
103

C

```

SUBROUTINE OUTPUT
COMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DINIT(2),GAINL,IWSTAT(5)
COMMON /DISPER/TSIGV(6,2),TSIGH(6,2),TMEANH(6,2),
1SSIGV(7,2),SSIGH(7,2)
COMMON /ENGR/SPDMMAX,HTMAX,HTMIN,TTS,TAA,TB,TFF
COMMON /DEF/TENGMX,SENGMX,SENGMN,TARTM,SARTM,TVEL,
1SVEL,DEFWTS(2)
COMMON /SUPEFT/GAMMA,DELTA

```

1

GRA04740  
GRA04750  
GRA04760  
GRA04770  
GRA04780  
GRA04790  
GRA04800  
GRA04810  
GRA04820  
GRA04830  
GRA04840  
GRA04850  
GRA04860  
GRA04870  
GRA04880  
GRA04890  
GRA04900  
GRA04910  
GRA04920  
GRA04930  
GRA04940  
GRA04950  
GRA04960  
GRA04970  
GRA04980  
GRA04990  
GRA05000  
GRA05010  
GRA05020  
GRA05030  
GRA05040  
GRA05050  
GRA05060  
GRA05070  
GRA05080  
GRA05090  
GRA05100  
GRA05110  
GRA05120  
GRA05130  
GRA05140  
GRA05150  
GRA05160  
GRA05170  
GRA05180  
GRA05190  
GRA05200  
GRA05210





GRA05220  
 GRA05230  
 GRA05240  
 GRA05250  
 GRA05260  
 GRA05270  
 GRA05280  
 GRA05290  
 GRA05300  
 GRA05310  
 GRA05320  
 GRA05330  
 GRA05340  
 GRA05350  
 GRA05360  
 GRA05370  
 GRA05380  
 GRA05390  
 GRA05400  
 GRA05410  
 GRA05420  
 GRA05430  
 GRA05440  
 GRA05450  
 GRA05460  
 GRA05470  
 GRA05480  
 GRA05490  
 GRA05500  
 GRA05510  
 GRA05520  
 GRA05530  
 GRA05540  
 GRA05550  
 GRA05560  
 GRA05570  
 GRA05580  
 GRA05590  
 GRA05600  
 GRA05610  
 GRA05620  
 GRA05630  
 GRA05640  
 GRA05650  
 GRA05660  
 GRA05670  
 GRA05680  
 GRA05690

```

C*** INPUT SUMMARY PRINTOUT
20 WRITE(6,20)
   FORMAT(1,1X,'AMPHIBIOUS ASSAULT INFORMATION')
22 FORMAT(1X,'INITIAL FORCE STRENGTH')
23 WRITE(6,23)
   FORMAT(1X,'WAVE',2X,'1',5X,'2',5X,'3',5X,'4',5X,'5')
24 WRITE(6,24)
   FORMAT(1X,'LVA',5(2X,F4.1))
21 WRITE(6,21)
   FORMAT(1X,'(DINIT(1),I=1,2)')
   DT = ,1X,F3.1,5X,'DS =',1X,F3.1)
25 WRITE(6,25)
   FORMAT(1X,'LVA ENGR SPECS')
26 WRITE(6,26)
   FORMAT(1X,'SPDMAX',2X,'SPDMIN',3X,'HITMAX',2X,'HITMIN',3X,'WID')
27 WRITE(6,27)
   FORMAT(2X,F5.2,3X,F5.2,3X,F5.2,3X,F4.2,2X,F4.2)
630 WRITE(6,630)
   FORMAT(1X,'DEFENSIVE TACTICAL PARAMETERS')
631 WRITE(6,631)
   FORMAT(10X,'RANGE',4X,'AIM-RELOAD',3X,'PROJECTILE')
632 WRITE(6,632)
   FORMAT(8X,'MAX',3X,'MIN',4X,'TIME',7X,'VELOCITY')
633 WRITE(6,633)
   FORMAT(1X,'TENG MX,TARTM,TVEL')
   TANK ,1X,F6.1,9X,F5.2,7X,F6.2)
634 WRITE(6,634)
   FORMAT(1X,'SENG MX,SENGMN,SARTM,SVEL')
   ATGM ,1X,F6.1,1X,F6.1,2X,F5.2,7X,F6.2)
50 WRITE(6,50)
   FORMAT(1X,'DEFENSIVE TACTICAL ALLOCATION WEIGHTS:',
1/ ,1X,'WAVE 1 =',F5.2,1X,'WAVE 2 =',F5.2)
100 WRITE(6,100)
   FORMAT(1X,'DEFENSIVE FORCE ATTRITION COEFFICIENTS')
101 WRITE(6,101)
   FORMAT(9X,'ALPHA*A',10X,'BETA*A')
102 WRITE(6,102)
   FORMAT(1X,'DT',6X,F7.5,9X,F7.5)
103 WRITE(6,103)
   FORMAT(1X,'DS',6X,F7.5,9X,F7.5)
104 WRITE(6,104)
   FORMAT(1X,'WB(1),WB(2)')
   WBETA(1) = ,F7.5,1X,'WBETA(2) =',F7.5)
105 WRITE(6,105)
   FORMAT(1X,'BREAKPOINT ASSUMPTION: 0.3*(TOTAL DEF FORCE)')
770 WRITE(6,770)
   FORMAT(1X,'DEFENDER ATTRITION LEVEL ALLOWING GROUND ATTACK',
1/ ,1X,F5.2,*(TOTAL DEFENDER FORCE))
   GAMMA,DELTA)
WRITE(6,771)

```



```

771 FORMAT(/IX,'ARTM SUP FACTOR=',F5.1,2X,'ERROR SUP FACTOR=',F5.1)
C*** DISPERSION DATA PRINTOUT
C
IDISP=1
IF (IDISP.EQ.0) RETURN
WRITE(6,601)
601 FORMAT(/IX,'DISPERSION DATA'/)
WRITE(6,602)
602 FORMAT(3X,'RANGE',2X,'TSIGV',2X,'RANGE',2X,'TSIGH',
12X,'RANGE',2X,'TMEANH')
DO 55 I=1,6
WRITE(6,603) TSIGV(I,1),TSIGV(I,2),TSIGH(I,1),TSIGH(I,2),
1TMEANH(I,1),TMEANH(I,2)
55 CONTINUE
603 FORMAT(1X,F7.1,2X,F5.1,1X,F7.1,1X,F5.1,1X,F7.1,1X,F5.1)
WRITE(6,604)
604 FORMAT(/3X,'RANGE',2X,'SSIGV',2X,'RANGE',2X,'SSIGH')
DO 56 I=1,7
WRITE(6,605) SSIGV(I,1),SSIGV(I,2),SSIGH(I,1),SSIGH(I,2)
56 CONTINUE
605 FORMAT(1X,F7.1,2X,F5.1,1X,F7.1,1X,F5.1)
WRITE(6,606)
606 FORMAT('1','THE AMPHIBIOUS ASSAULT PHASE BEGINS'///)
RETURN
END

C
SUBROUTINE PHIT(RANGE,W,H,IWPN,SUPFAC,PRHIT)
COMMON /AMPH/IL(5),WB(2),A(2),B(2),ITE,ISE,RD,WVINT(5),WID,
1TBW,DINIT(2),GAINL,IWSTAT(5)
COMMON /DISPER/TSIGV(6,2),TSIGH(6,2),TMEANH(6,2),
1SSIGV(7,2),SSIGH(7,2)
COMMON /SUPEFT/GAMMA,DELTA

C
PI=ARCCOS(-1.0)
IF(RANGE.LT.25.) STOP
IF(IWPN.EQ.1) GO TO 50
C ATGM FIRING DATA COMPUTATIONS
WMEANH=0.0
WMEANV=0.0
CALL INTRP(SSIGV,RANGE,WSIGV,7)
CALL INTRP(SSIGV,RANGE,WSIGV,7)
C TANK FIRING DATA COMPUTATIONS
50 WMEANH=0.0
CALL INTRP(TMEANH,RANGE,WMEANH,6)
CALL INTRP(TSIGV,RANGE,WSIGV,6)
CALL INTRP(TSIGH,RANGE,WSIGV,6)
C CONVERSION TO MILS

```



```

100 Z=ARSIN(H/RANGE)
    WSIGV=WSIGV*(1.+DELTA*SUPFAC)
    WSIGH=WSIGH*(1.+DELTA*SUPFAC)
    TGTW=(Z*6400.0)/(2.0*PI)
    TGTW=(ARSIN(W/RANGE))*(6400.0/(2.0*PI))

C INSTITUTE NORMALITY ASSUMPTIONS TO CCMPUTE HORIZONTAL
C AND VERTICAL HIT PROBABILITIES
C
    C=-1.0*SQRT(1./2.)
    HOR1=((TGTW/2.)-WMEANH)/WSIGH
    HOR2=((-1.0*TGTW)/2.0)-WMEANH)/WSIGH
    PHITX=1.0
    IF(ABS(HOR1).GT.8.) GO TO 810
    PHITX=0.5*(ERFC(C*HOR1)-ERFC(C*HOR2))
810 VER1=((TGTW/2.)-WMEANH)/WSIGV
    VER2=((-1.0*TGTW)/2.0)-WMEANH)/WSIGV
    PHITY=1.0
    IF(ABS(VER1).GT.8.) GO TO 820
    PHITY=0.5*(ERFC(C*VER1)-ERFC(C*VER2))
820 PRHIT=PHITX*PHITY
    RETURN
    END

C
C SUBROUTINE INTRP(X,ARG,VAL,N)
C DIMENSION X(N,2)
C WRITE(6,777) ARG
C 777 FORMAT(1X,'ARG**'= ,F10.3)
    IF(ARG.LT.X(1,1)) GO TO 500
    DO 50 I=1,N
    IF(ARG.GT.X(I+1,1)) GO TO 50
    DIFF=X(I+1,1)-X(I,1)
    DELTA=ARG-X(I,1)
    VAL=X(I,2)+(DELTA/DIFF)*(X(I+1,2)-X(I,2))
    RETURN
50 CONTINUE
    IF(ARG.GT.X(N,1)) GO TO 600
    VAL=X(N,2)
    RETURN
600 WRITE(6,601)
601 FORMAT(1X,'ERROR IN INTRP ARG.GT.X(N,2)')
    STOP
500 WRITE(6,501)
501 FORMAT(1X,'ERROR IN INTRP ARG.LT.X(1,1)')
    STOP
    END

C SUBROUTINE RATE(RANGE,SPEED,IWPN,SUPFAC,ROF)

```



```

COMMON /DEF/ TENG MX, SENG MX, SENG MN, TART M, SART M, TVEL, SVEL
COMMON /SUF/ FT/ GAM MA, DELTA
ROF=0.0
IF(RANGE.LT.25.) RETURN
IF(IWPN.EQ.2) GO TO 500
IF(RANGE.GT.TENG MX) RETURN
TRTM=TART M*(1.0+GAM MA*SUPFAC)
DT=TRTM+RANGE/(TVEL+SPEED)
ROF=1.0/DT
RETURN
500 IF(RANGE.GT.SENG MX) RETURN
IF(RANGE.LT.SENG MN) RETURN
SRTM=SART M*(1.0+GAM MA*SUPFAC)
DT=SRTM+RANGE/(SVEL+SPEED)
ROF=1.0/DT
RETURN
END

C
FUNCTION SPD(T)
COMMON /ENGR/ SPD MAX, SPD MIN, HT MAX, HT MIN, TTS, TAA, TB, TFF
IF(T.GT.TAA) GO TO 50
SPD=SPD MAX
RETURN
50 IF(T.GT.TB) GO TO 100
SPD=SPD MIN+((TB-T)/TTS)*(SPD MAX-SPD MIN)
RETURN
100 SPD=SPD MIN
RETURN
END

C
FUNCTION HT(T)
COMMON /ENGR/ SPD MAX, SPD MIN, HT MAX, HT MIN, TTS, TAA, TB, TFF
IF(T.GT.TAA) GO TO 50
HT=HT MAX
RETURN
50 IF(T.GT.TB) GO TO 100
HT=HT MIN+((TB-T)/TTS)*(HT MAX-HT MIN)
RETURN
100 HT=HT MIN
RETURN
END

C
FUNCTION RNG(T)
COMMON /AMPH/ IL(5), WB(2), A(2), B(2), ITE, ISE, RD, WV INT(5), WID,
1TBW, DIN IT(2), GAIN L, IW STAT(5)
COMMON /ENGR/ SPD MAX, SPD MIN, HT MAX, HT MIN, TTS, TAA, TB, TFF
IF(T.GT.TAA) GO TO 50

```

GRA06660  
 GRA06670  
 GRA06680  
 GRA06690  
 GRA06700  
 GRA06710  
 GRA06720  
 GRA06730  
 GRA06740  
 GRA06750  
 GRA06760  
 GRA06770  
 GRA06780  
 GRA06790  
 GRA06800  
 GRA06810  
 GRA06820  
 GRA06830  
 GRA06840  
 GRA06850  
 GRA06860  
 GRA06870  
 GRA06880  
 GRA06890  
 GRA06900  
 GRA06910  
 GRA06920  
 GRA06930  
 GRA06940  
 GRA06950  
 GRA06960  
 GRA06970  
 GRA06980  
 GRA06990  
 GRA07000  
 GRA07010  
 GRA07020  
 GRA07030  
 GRA07040  
 GRA07050  
 GRA07060  
 GRA07070  
 GRA07080  
 GRA07090  
 GRA07100  
 GRA07110  
 GRA07120  
 GRA07130





```

RNG=5000.0-(SPDMA X*T)
RETURN
50 IF(T.GT.TB) GO TO 100
RNG=RD-0.5*(T-TAA)*(SPDMAX+SPD(T))
RETURN
100 IF(RNG=RD-((TB-TAA)/2.0)*((SPDMIN+SPDMAX))-(T-TB)*SPDMIN)
IF(RNG.LT.75.) RNG=0.0
RETURN
END

SUBROUTINE GROUND(GATM)
COMMON /GRP1/ IPRDIR(6), ISECWD(6), MVTDIR(6), X(6), Y(6), SPD(6)
COMMON /GRP2/ TA(2), TM(2), TF1(2), TF2(2), TF3(2),
1P(2,6), PHH(2,6), PHM(2,6), PKH(2,6), TF(2)
COMMON /GRP3/ NBU, NRU, FL(6), FO(6), NOI(3), XIC(3,200), YIC(3,200),
1IDIR(3,200), AVSP, ISPD
COMMON /GRP4/ TPOL(6), LOT(6,6), ROT(6,6)
COMMON /GRP5/ XC(100), YC(100), PEAK(100), SX(100), SY(100), RHO(100)
COMMON /HILLS/ SCALE(100), TWORHO(100), TWOSCL(100), BASE
COMMON /HILLS/ NHILLS
COMMON /COVER/ CX(150), CPEAK(150), CPXX(150), CPYY(150)
COMMON /COVER/ CPXY(150), NCVELS
COMMON /COUNTR/ KH, KHW, KV, KN, KGRS, KELL, KINT
COMMON /GRID/ LST(10,10), NHL(10,10), LSTH(450), KHREP(100), KTREP
COMMON /GRID/ LSTC(10,10), NC(10,10), LISTC(400), KCREP(150)
COMMON /GRP6/ ALPHA(6)
COMMON /GRP7/ XA(6), YA(6), IMOVE(6)
INITIALIZATION.

BL=0.0
RL=0.0
MP=0
PAI=3.14159
ZL=.00001

READ TERRAIN DATA FOR LINE OF SIGHT
CHECK FOR STOCHASTIC OR DETERMINISTIC ATTRITION
ITRIT-ATTRITION MOD 1=DETERMINISTIC
0=STOCHASTIC
IS-SEED NUMBER

```

CC

CC

CC  
CC  
CC  
CC  
CC



```

130 READ(9,130) ITRIT,IS
   FORMAT(1,1X,I5)
   DO 132 I=1,6
   CALL LRND(1,IS,YRAN,1,1,0)
   ALPHA(1)=(-2.*YRAN**2)+(2.*YRAN+.3)
   WRITE(6,799) YRAN,ALPHA(1)
C
132 CONTINUE
799 FORMAT(2X,'YRAN,ALPHA',F10.5,2X,F10.5)
C
C READ IN NUMBER OF ATTACK AND DEFENSE UNITS
C
200 READ(9,200) NBU,NRU
   FORMAT(12,1X,I2)
C
C INITIALIZE WEAPON SIZES
C
   SIZETK=2.5
   SIZETW=2.5
C
C READ IN EFFECTIVE WEAPON RANGES
C
102 READ(9,102) RMINTK,RMXTK,RMINTW,RMXTW
   FORMAT(F6.1,1X,F6.1,1X,F6.1,1X,F6.1,1X)
C
C INITIALIZE PM,RF,TOWER,TNKFR AND NOD
C
   PM=.352
   RF=.5
   TOWER=.03
   TNKFR=.1
   NOD=2
   DO 101 I=1,NRU
   NOI(I)=125
   CONTINUE
101 K=NRU+1
   L=NRU+NBU
   DO 111 I=1,L
   II(I)=0
   CONTINUE
111 CONTINUE
C
C READ IN FORCE LEVELS OF EACH ATTACK UNIT
C
103 READ(9,103) (FL(I),I=1,NRU)
   FORMAT(3(F3.1,1X))
C
C CHECK FOR TYPE OF ROUTE DETERMINATION
C
   READ(9,106) IRTE,ISPD

```

```

GRA07620
GRA07630
GRA07640
GRA07650
GRA07660
GRA07670
GRA07680
GRA07690
GRA07700
GRA07710
GRA07720
GRA07730
GRA07740
GRA07750
GRA07760
GRA07770
GRA07780
GRA07790
GRA07800
GRA07810
GRA07820
GRA07830
GRA07840
GRA07850
GRA07860
GRA07870
GRA07880
GRA07890
GRA07900
GRA07910
GRA07920
GRA07930
GRA07940
GRA07950
GRA07960
GRA07970
GRA07980
GRA07990
GRA08000
GRA08010
GRA08020
GRA08030
GRA08040
GRA08050
GRA08060
GRA08070
GRA08080
GRA08090

```



GRA08100  
GRA08110  
GRA08120  
GRA08130  
GRA08140  
GRA08150  
GRA08160  
GRA08170  
GRA08180  
GRA08190  
GRA08200  
GRA08210  
GRA08220  
GRA08230  
GRA08240  
GRA08250  
GRA08260  
GRA08270  
GRA08280  
GRA08290  
GRA08300  
GRA08310  
GRA08320  
GRA08330  
GRA08340  
GRA08350  
GRA08360  
GRA08370  
GRA08380  
GRA08390  
GRA08400  
GRA08410  
GRA08420  
GRA08430  
GRA08440  
GRA08450  
GRA08460  
GRA08470  
GRA08480  
GRA08490  
GRA08500  
GRA08510  
GRA08520  
GRA08530  
GRA08540  
GRA08550  
GRA08560  
GRA08570

```

106 FORMAT(11,1X,11)
   IF(I SPD.EQ.1) AVSP=9.0
   IF(I SPD.EQ.1) DST=40.232
   IF(I SPD.EQ.2) AVSP=12.0
   IF(I SPD.EQ.2) DST=53.643
   IF(I SPD.EQ.3) AVSP=15.0
   IF(I SPD.EQ.3) DST=67.053
   IF(I SPD.EQ.4) AVSP=18.0
   IF(I SPD.EQ.4) DST=80.463

C
C  READ IN INITIAL ATTACK UNIT'S LOCATIONS
C
      DC 6 I=1,NRU
      READ(9,107) XIC(I,1),YIC(I,1)
107 6 FORMAT(F6.1,1X,F6.1)
      CONTINUE
      DO 2 I=1,NRU
      DO 2 J=2,125
        YIC(I,J)=YIC(I,J-1)+DST*(J-1)
        XIC(I,J)=XIC(I,J-1)+DST*(J-1)
        IDIR(I,J)=0
      2 CONTINUE
      GO TO 109
108 CALL ROUTE
109 SUMRO=0.0
      DO 3 I=1,NRU
        FO(I)=FL(I)
        SUMRO=SUMRO+FO(I)
        X(I)=XIC(I,1)
        Y(I)=YIC(I,1)
        MVTDIR(I)=IDIR(I,1)
        SPD(I)=AVSP
        IUSTAT(I)=0
        IPRDIR(I)=IDIR(I,1)
        ISECWD(I)=120
        NF(I)=1
        II(I)=1
      3 CONTINUE

C
C  READ IN DEFENSE UNIT'S LOCATIONS
C
      SUMBO=0.0
      DO 4 I=K,L
104 4 I=K,L
      READ(9,104) X(I),Y(I),FL(I),IPRDIR(I),ISECWD(I)
      FORMAT(F6.1,1X,F6.1,1X,F3.1,1X,I3,1X,I3)
      FO(I)=FL(I)
      SUMBO=SUMBO+FO(I)

```



```

MVTDIR(I)=0
SPD(I)=0.0
IUSTAT(I)=0
IMOVE(I)=0
4 CONTINUE

C CHECK FOR ALTERNATE DEFENSE POSITIONS AND READ IN IF WANTED
C
      READ(9,400) IALT, BREAK, ITEM
400  FORMAT(I1,1X,F6.1,1X,I2)
      IF(IALT.EQ.1) GO TO 401
      DO 402 I=K,L
402  READ(9,107) XA(I),YA(I)
401  CONTINUE
      TA(1)=20.
      TI(1)=8.
      TH(1)=8.
      TM(1)=10.
      TF1(1)=1.
      TF2(1)=1.
      TF3(1)=1.
      TA(2)=20.
      TI(2)=8.
      TH(2)=8.
      TM(2)=15.
      TF1(2)=10.
      TF2(2)=12.
      TF3(2)=15.

C READ IN HIT AND KILL PROBABILITIES
C
      DO 5 I=1,2
      DO 514 J=1,6
      READ(9,515) P(I,J),PHH(I,J),PHM(I,J),PKH(I,J)
515  FORMAT(4(F4.2,1X))
514  CONTINUE
      PTT(1,1)=1.0
      PTT(1,2)=0.8
      PTT(2,2)=0.2
      PTT(1,3)=0.8
      PTT(2,3)=0.15
      DC 31 I=1,NRU
      DO 31 J=K,L
      NLOSC(I,J)=0
      NLOSC(J,I)=0

```

```

GRA08580
GRA08590
GRA08600
GRA08610
GRA08620
GRA08630
GRA08640
GRA08650
GRA08660
GRA08670
GRA08680
GRA08690
GRA08700
GRA08710
GRA08720
GRA08730
GRA08740
GRA08750
GRA08760
GRA08770
GRA08780
GRA08790
GRA08800
GRA08810
GRA08820
GRA08830
GRA08840
GRA08850
GRA08860
GRA08870
GRA08880
GRA08890
GRA08900
GRA08910
GRA08920
GRA08930
GRA08940
GRA08950
GRA08960
GRA08970
GRA08980
GRA08990
GRA09000
GRA09010
GRA09020
GRA09030
GRA09040
GRA09050

```





GRA09060  
GRA09070  
GRA09080  
GRA09090  
GRA09100  
GRA09110  
GRA09120  
GRA09130  
GRA09140  
GRA09150  
GRA09160  
GRA09170  
GRA09180  
GRA09190  
GRA09200  
GRA09210  
GRA09220  
GRA09230  
GRA09240  
GRA09250  
GRA09260  
GRA09270  
GRA09280  
GRA09290  
GRA09300  
GRA09310  
GRA09320  
GRA09330  
GRA09340  
GRA09350  
GRA09360  
GRA09370  
GRA09380  
GRA09390  
GRA09400  
GRA09410  
GRA09420  
GRA09430  
GRA09440  
GRA09450  
GRA09460  
GRA09470  
GRA09480  
GRA09490  
GRA09500  
GRA09510  
GRA09520  
GRA09530

```

Q(I,J)=1.0
Q(J,I)=1.0
VISFR(I,J)=0.0
VISFR(J,I)=0.0
31 CONTINUE
IC=1

C PRINT INITIAL BATTLE INFORMATION
C
C
599 WRITE(6,599)
600 FORMAT(1,1X,'INITIAL GROUND COMBAT INFORMATION')
600 WRITE(6,600)
600 FORMAT(1X,'UNIT',7X,'X',8X,'Y',4X,'FORCE LEVEL')
DO 601 I=1,L
602 WRITE(6,602) I,X(I),Y(I),FL(I)
602 FORMAT(1X,I3,3X,F7.1,2X,F7.1,7X,F3.1)
601 CONTINUE
IF(IIRIT.EQ.1) GO TO 603
WRITE(6,604)
604 FORMAT(1X,'ATTRITION IS STOCHASTIC')
GO TO 605
603 WRITE(6,606)
606 FORMAT(1X,'ATTRITION IS DETERMINISTIC')
606 IF(IRTE.EQ.0) GO TO 607
605 WRITE(6,608)
608 FCRMAT(1X,'ROUTES DETERMINED BY USER')
607 WRITE(6,609) AVSP
609 FORMAT(1X,'ATTACK VEHICLE SPEED IS ',F4.1)
610 WRITE(6,610) BREAK
610 FORMAT(1X,'BREAKPOINT DISTANCE IS ',F6.1)
IF(IALT.EQ.0) GO TO 615
WRITE(6,620)
620 FORMAT(1X,'DEFENDER WILL NOT MOVE TO ALTERNATE POSITIONS')
GO TO 625
615 WRITE(6,630)
630 FORMAT(1X,'DEFENDER WILL MOVE TO ALTERNATE POSITIONS',1X,
1,'ALTERNATE POSITIONS ARE:',1X,'UNIT',5X,'X',8X,'Y')
DO 635 I=K,L
640 WRITE(6,640) I,XA(I),YA(I)
635 FORMAT(1X,I3,3X,F7.1,2X,F7.1)
635 CONTINUE
625 IRAN=500
645 WRITE(6,645)
645 FORMAT(1X,'ATK KILL PROBABILITIES',1X,'RANGE',4X,'P',
14X,'PHM',3X,'PHM',3X,'PKH')
DO 650 I=1,6
650 WRITE(6,655) IRAN,P(1,I),PHH(1,I),PHM(1,I),PKH(1,I)
655 FORMAT(2X,I4,4(2X,F4.2))

```



GRA09540  
GRA09550  
GRA09560  
GRA09570  
GRA09580  
GRA09590  
GRA09600  
GRA09610  
GRA09620  
GRA09630  
GRA09640  
GRA09650  
GRA09660  
GRA09670  
GRA09680  
GRA09690  
GRA09700  
GRA09710  
GRA09720  
GRA09730  
GRA09740  
GRA09750  
GRA09760  
GRA09770  
GRA09780  
GRA09790  
GRA09800  
GRA09810  
GRA09820  
GRA09830  
GRA09840  
GRA09850  
GRA09860  
GRA09870  
GRA09880  
GRA09890  
GRA09900  
GRA09910  
GRA09920  
GRA09930  
GRA09940  
GRA09950  
GRA09960  
GRA09970  
GRA09980  
GRA09990  
GRA10000  
GRA10010

```

650  IRAN=IRAN+500
      CONTINUE
      WRITE(6,660)
660  FORMAT(/4X,'DEF. KILL PROBABILITIES'/1X,'RANGE',4X,'P',
14X,'PHH',3X,'PHM',3X,'PKH')
      DO 665 I=1,6
        WRITE(6,655) IRAN,P(2,I),PHH(2,I),PHM(2,I),PKH(2,I)
665  IRAN=IRAN+500
      CONTINUE
      WRITE(6,670)
670  FORMAT(11,10X,'BATTLE BEGINS',/)
      C  UPDATE LOCATION OF RED UNITS.
      DISMAX=5000.0
67  DO 9 I=1,NRU
      IF(IUSTAT(I).EQ.2) GOTO 9
      IF(IUSTAT(I).EQ.0) GOTO 76
      NF(I)=NF(I)+1
      IF(NF(I).LT.NOD) GOTO 9
      NF(I)=1
76  DO 11 J = 1, NRU
      IF (J.EQ. 1) GO TO 11
      IF (IUSTAT(J).EQ. 2) GO TO 11
      DIST = X(I) - X(J)
      IF (DIST.GT. DISMAX) GO TO 9
11  CONTINUE
      II(I) = II(I) + 1
      K7=II(I)
      X(I)=XIC(I,K7)
      Y(I)=YIC(I,K7)
      MVIDIR(I)=IDIR(I,K7)
      IPRDIR(I)=IDIR(I,K7)
      WRITE(6,666) I,X(I),Y(I),MVIDIR(I),IPRDIR(I)
      C 666 FORMAT(/,1X,I3,1X,F10.5,2X,F10.5,2X,I10,2X,I10,/)
      C 9 CONTINUE
      C  LINE--OF-SIGHT CHECK BETWEEN UNITS AND TARGETS SELECTION
      DO 17 J=K,L
      NT(J)=0
17  CONTINUE
      DO 12 I=1,NRU
      NT(I)=0
      IF(IUSTAT(I).EQ.2) GOTO 12
      DO 16 J=K,L
      IF(IUSTAT(J).EQ.2.OR.IUSTAT(J).EQ.3) GO TO 16
      XX1=X(I)

```



GRA10020  
GRA10030  
GRA10040  
GRA10050  
GRA10060  
GRA10070  
GRA10080  
GRA10090  
GRA10100  
GRA10110  
GRA10120  
GRA10130  
GRA10140  
GRA10150  
GRA10160  
GRA10170  
GRA10180  
GRA10190  
GRA10200  
GRA10210  
GRA10220  
GRA10230  
GRA10240  
GRA10250  
GRA10260  
GRA10270  
GRA10280  
GRA10290  
GRA10300  
GRA10310  
GRA10320  
GRA10330  
GRA10340  
GRA10350  
GRA10360  
GRA10370  
GRA10380  
GRA10390  
GRA10400  
GRA10410  
GRA10420  
GRA10430  
GRA10440  
GRA10450  
GRA10460  
GRA10470  
GRA10480  
GRA10490

```

133 YY1=Y(I)
    CALL ELLEV(XX1,YY1,TMACI)
    XX2=X(J)
    YY2=Y(J)
    CALL ELLEV(XX2,YY2,TMACJ)
    LATOB=1
    LBTOA=1
    WRITE(6,675) XX1,YY1,TMACI,XX2,YY2,TMACJ
    FORMAT(IX,PRELOS,IX,6(F10.5,IX))
    CALL LOS(XX1,YY1,TMACI,0.0,SIZETK,XX2,YY2,TMACJ,0.0,SIZETW,
1  LATOB,LBTOA,VISFRA,VISFRB)
    VISFR(I,J)=VISFRA
    VISFR(J,I)=VISFRB
    IF(VISFRA.GT.ZL) GOTO 18
    LOST(I,J)=0
    LOST(J,I)=0
    NLOSC(I,J)=NLOSC(I,J)+1
    NLOSC(J,I)=NLOSC(I,J)
    GOTO 16
18  LOST(I,J)=1
    LOST(J,I)=1
    NLOSC(I,J)=0
    NLOSC(J,I)=0
    RANGE=SQRT((X(I)-X(J))**2+(Y(I)-Y(J))**2)
    IF(RANGE.LT.RMINTK.OR.RANGE.GT.RMXTK)GOTO 20
    IF(Q(I,J).EQ.1.0) GOTO 20
    IUSTAT(I)=1
    NT(I)=NT(I)+1
    M=NT(I)
    LOT(I,M)=J
    RCT(I,M)=RANGE
    IF(M.EQ.1) GOTO 20
    CALL SORT(I,M)
20  IF(RANGE.LT.RMINTW.OR.RANGE.GT.RMXTW) GOTO 16
    IF(Q(J,I).EQ.1.0) GOTO 16
    IUSTAT(J)=1
    NT(J)=NT(J)+1
    M=NT(J)
    LOT(J,M)=I
    RCT(J,M)=RANGE
    IF(M.EQ.1) GOTO 16
    CALL SORT(J,M)
16  CONTINUE
12  DO 25 I=1,NRU
    IF(IUSTAT(I).EQ.2) GOTO 25
    IF(NT(I).NE.0) GOTO 25
    IUSTAT(I)=0

```









GRA10980  
GRA10990  
GRA11000  
GRA11010  
GRA11020  
GRA11030  
GRA11040  
GRA11050  
GRA11060  
GRA11070  
GRA11080  
GRA11090  
GRA11100  
GRA11110  
GRA11120  
GRA11130  
GRA11140  
GRA11150  
GRA11160  
GRA11170  
GRA11180  
GRA11190  
GRA11200  
GRA11210  
GRA11220  
GRA11230  
GRA11240  
GRA11250  
GRA11260  
GRA11270  
GRA11280  
GRA11290  
GRA11300  
GRA11310  
GRA11320  
GRA11330  
GRA11340  
GRA11350  
GRA11360  
GRA11370  
GRA11380  
GRA11390  
GRA11400  
GRA11410  
GRA11420  
GRA11430  
GRA11440  
GRA11450

```

IF(ANG.GT.AA) GOTO 24
PROP=PROP+PTT(I1,N5)
24 CONTINUE
IF(PROP.EQ.0.0) GOTO 34
IF(INT(J).GT.0)GOTO 36
CALL LAMDA(I,J,PCTVIS,DETRAT,PSUBK)
DETRAT=DETRAT*RF
QV=EXP(-(FL(I)*PROP*DETRAT*DELT*FL(J)))
Q(I,J)=Q(I,J)*QV
GOTO 19
36 Q(I,J)=0.0
GOTO 19
34 IF(IAA.EQ.1) GOTO 19
Q(I,J)=1.0
GOTO 19
15 IF(NLOSC(I,J).LE.3) GOTO 19
19 Q(I,J)=1.0
CONTINUE
14 CONTINUE
IF(IAA.EQ.K) GOTO 38
FR=TNKFR
IAA=K
IBB=L
IBCC=1
IDD=NRU
OP=1.0
GOTO 37

FIRE ALLOCATION.

38 DO 28 I=1,L
28 NA(I)=0
DO 26 I=1,L
IF(IUSTAT(I).EQ.2.OR.IUSTAT(I).EQ.3) GO TO 26
IF(INT(I).EQ.0) GOTO 26
DC 27 J=1,3
APOA(I,J)=0.0
27 CONTINUE
IF(INT(I).EQ.1) GOTO 78
IF(INT(I).EQ.2) GOTO 29
NOT=3
MM1=LOT(I,1)
MM2=LOT(I,2)
MM3=LOT(I,3)
PROB=(1.0-Q(I,MM1))*Q(I,MM2)*Q(I,MM3)
APOA(I,1)=APOA(I,1)+PTT(I,1)*PROB
PROB=Q(I,MM1)*(1.0-Q(I,MM2))*Q(I,MM3)
APOA(I,2)=APOA(I,2)+PTT(I,1)*PROB

```

C  
C  
C



GRA11460  
 GRA11470  
 GRA11480  
 GRA11490  
 GRA11500  
 GRA11510  
 GRA11520  
 GRA11530  
 GRA11540  
 GRA11550  
 GRA11560  
 GRA11570  
 GRA11580  
 GRA11590  
 GRA11600  
 GRA11610  
 GRA11620  
 GRA11630  
 GRA11640  
 GRA11650  
 GRA11660  
 GRA11670  
 GRA11680  
 GRA11690  
 GRA11700  
 GRA11710  
 GRA11720  
 GRA11730  
 GRA11740  
 GRA11750  
 GRA11760  
 GRA11770  
 GRA11780  
 GRA11790  
 GRA11800  
 GRA11810  
 GRA11820  
 GRA11830  
 GRA11840  
 GRA11850  
 GRA11860  
 GRA11870  
 GRA11880  
 GRA11890  
 GRA11900  
 GRA11910  
 GRA11920  
 GRA11930

```

PROB=Q(I,MM1)*Q(I,MM2)*(1.0-Q(I,MM3))
APOA(I,3)=APOA(I,1)+PTT(1,1)*PROB
PROB=(1.0-Q(I,MM1))*(1.0-Q(I,MM2))*Q(I,MM3)
APOA(I,1)=APOA(I,1)+PTT(1,2)*PROB
APOA(I,2)=APOA(I,2)+PTT(2,2)*PROB
PROB=(1.0-Q(I,MM1))*Q(I,MM2)*(1.0-Q(I,MM3))
APOA(I,1)=APOA(I,1)+PTT(1,2)*PROB
APOA(I,3)=APOA(I,3)+PTT(2,2)*PROB
PROB=Q(I,MM1)*(1.0-Q(I,MM2))*(1.0-Q(I,MM3))
APOA(I,2)=APOA(I,2)+PTT(1,2)*PROB
APOA(I,3)=APOA(I,3)+PTT(2,2)*PROB
PROB=(1.0-Q(I,MM1))*(1.0-Q(I,MM2))*(1.0-Q(I,MM3))
APOA(I,1)=APOA(I,1)+PTT(1,3)*PROB
APOA(I,2)=APOA(I,2)+PTT(2,3)*PROB
APOA(I,3)=APOA(I,3)+PTT(3,3)*PROB
DO 44 J=1,NOT
KK=LOT(I,J)
NA(KK)=NA(KK)+1
IN=NA(KK)
LOA(KK,IN)=I
POA(KK,IN)=APOA(I,J)
CONTINUE
44 GOTO 26
29 NOT=2
MM1=LOT(I,1)
MM2=LOT(I,2)
PROB=(1.0-Q(I,MM1))*Q(I,MM2)
APOA(I,1)=APOA(I,1)+PTT(1,1)*PROB
PROB=Q(I,MM1)*(1.0-Q(I,MM2))
APOA(I,2)=APOA(I,2)+PTT(1,1)*PROB
PROB=(1.0-Q(I,MM1))*Q(I,MM2)
APOA(I,1)=APOA(I,1)+PTT(1,2)*PROB
APOA(I,2)=APOA(I,2)+PTT(2,2)*PROB
GOTO 30
78 NOT=1
MM1=LOT(I,1)
PROB=1.0-Q(I,MM1)
APOA(I,1)=APOA(I,1)+PTT(1,1)*PROB
GOTO 30
26 CONTINUE
ATTRITION.
SUMR=0.0
SUMB=0.0
DO 40 I=1,L
IF(IUSTAT(I).EQ.2.OR.IUSTAT(I).EQ.3) GO TO 40
  
```

C  
 C  
 C  
 C



```

M6=NA(I)
SUM=0.0
IF(M6.EQ.0) GOTO 47
DO 41 J=1,M6
M7=LOA(I,J)
IF(M7.LT.K) GOTO 42
I TYPE=2
GOTO 43
42 I TYPE=1
43 RANGE=SQR T((X(I)-X(M7))**2+(Y(I)-Y(M7))**2)
IF (I TRIT.EQ.1) GO TO 131
CALL STOCH(I TYPE,RANGE,AJI)
GO TO 5000
131 CALL ET K(I TYPE,RANGE,T)
AJI=1.0/T
SUM=SUM+AJI*FL(M7)*POA(I,J)*DEL T
CONTINUE
41 OFL(I)=FL(I)
47 FL(I)=FL(I)-SUM
IF(FL(I).GT.ZL) GOTO 46
FL(I)=0.0
IUSTAT(I)=2
IF(I.LT.K) GOTO 60
SUMB=SUMB+FL(I)
TPOL(I)=(FO(I)-FL(I))/FO(I)
GO TO 40
60 SUMR=SUMR+FL(I)
TPOL(I)=(FO(I)-FL(I))/FO(I)
40 CONTINUE

PRINT AND CHECK FOR BATILE DETERMINATION.

C
C
C
C
ITIME=IC*10
DO 57 I=K,L
IF(IUSTAT(I).EQ.2) GO TO 57
DO 58 J=1,NRU
IF(IUSTAT(J).EQ.2) GO TO 58
CHECK=X(I)-X(J)
AVD=SQR T((X(I)-X(J))**2+(Y(I)-Y(J))**2)
IF(AVD.LT.BREAK.OR.CHECK.LT.50.) GO TO 250
58 CONTINUE
57 GO TO 99

C
C
C
C
COMPLETE ATTACK UNIT'S MOVE
250 DO 251 I=K,L

```



GRA12420  
GRA12430  
GRA12440  
GRA12450  
GRA12460  
GRA12470  
GRA12480  
GRA12490  
GRA12500  
GRA12510  
GRA12520  
GRA12530  
GRA12540  
GRA12550  
GRA12560  
GRA12570  
GRA12580  
GRA12590  
GRA12600  
GRA12610  
GRA12620  
GRA12630  
GRA12640  
GRA12650  
GRA12660  
GRA12670  
GRA12680  
GRA12690  
GRA12700  
GRA12710  
GRA12720  
GRA12730  
GRA12740  
GRA12750  
GRA12760  
GRA12770  
GRA12780  
GRA12790  
GRA12800  
GRA12810  
GRA12820  
GRA12830  
GRA12840  
GRA12850  
GRA12860  
GRA12870  
GRA12880  
GRA12890

```

IF(IALT.EQ.1.OR.IMOVE(I).EQ.ITEM) GO TO 6000
IF(IUSTAT(I).EQ.0) IUSTAT(I)=3
IMOVE(I)=IMOVE(I)+1
IF(IMOVE(I).LT.ITEM) GO TO 251
X(I)=XA(I)
Y(I)=YA(I)
IF(IUSTAT(I).EQ.3) IUSTAT(I)=0
251 CONTINUE
99 IITIME=IITIME+IFIX(GATM)
WRITE(6,112) IITIME
112 FORMAT(//1X,'TIME=',I4,1X,'SECONDS'//)
113 WRITE(6,113)
113 FORMAT(1X,'UNIT',5X,'X',8X,'Y',5X,'FORCE LEVEL',2X,'STATUS',
12X,'LOST-PCT',2X,'TARGETS')
DO 59 I=1,L
N6=NT(I)
IF(N6.NE.0) GO TO 48
WRITE(6,264) I,X(I),Y(I),FL(I),IUSTAT(I),TPOL(I)
264 FORMAT(3X,I1,3X,F7.1,2X,F7.1,6X,F3.1,9X,I1,6X,F5.3)
48 GO TO 59
WRITE(6,114) I,X(I),Y(I),FL(I),IUSTAT(I),TPOL(I),
1(LOT(I),J),J=1,N6)
114 FORMAT(3X,I1,3X,F7.1,2X,F7.1,6X,F3.1,9X,I1,6X,F5.3,3X,I1,1X)
59 CONTINUE
C
C CHECK FOR BATTLE TERMINATION.
C
IOT=0
DO 53 I=1,NRU
IF(FL(I).EQ.0.0) GOTO 53
IOT=I
53 CONTINUE
IF(IOT.EQ.1) GOTO 54
WRITE(6,117)
117 FORMAT(1X,'*ATTACK FORCE IS ELIMINATED. END OF BATTLE.')
54 GOTO 66
IOT=0
DO 55 I=K,L
IF(FL(I).EQ.0.0) GOTO 55
IOT=I
55 CONTINUE
IF(IOT.EQ.1) GOTO 65
WRITE(6,118)
118 FORMAT(1X,'*DEFENSE FORCE IS ELIMINATED. END OF BATTLE.')
6000 WRITE(6,119)
119 FORMAT(1X,'*DISTANCE BETWEEN FORCES IS TOO CLOSE. END OF BATTLE
1.)

```





GRA12900  
 GRA12910  
 GRA12920  
 GRA12930  
 GRA12940  
 GRA12950  
 GRA12960  
 GRA12970  
 GRA12980  
 GRA12990  
 GRA13000  
 GRA13010  
 GRA13020  
 GRA13030  
 GRA13040  
 GRA13050  
 GRA13060  
 GRA13070  
 GRA13080  
 GRA13090  
 GRA13100  
 GRA13110  
 GRA13120  
 GRA13130  
 GRA13140  
 GRA13150  
 GRA13160  
 GRA13170  
 GRA13180  
 GRA13190  
 GRA13200  
 GRA13210  
 GRA13220  
 GRA13230  
 GRA13240  
 GRA13250  
 GRA13260  
 GRA13270  
 GRA13280  
 GRA13290  
 GRA13300  
 GRA13310  
 GRA13320  
 GRA13330  
 GRA13340  
 GRA13350  
 GRA13360  
 GRA13370

```

65 GO TO 66
   IC=IC+1
66 GO TO 67
   RETURN
   END

C SUBROUTINE SETUP
C SUBROUTINE SETUP IS USED TO READ IN THE TERRAIN DATA AND
C CREATE PARAMETRIC TERRAIN. THIS TERRAIN DATA WILL BE USED
C WHEN COMPUTING LINE-OF-SIGHT BETWEEN TARGETS AND OBSERVERS
C AS WELL AS PROVIDING A GRID SYSTEM FOR UNIT LOCATIONS AND
C MOVEMENT.

COMMON /HILLS/ XC(100),YC(100),PEAK(100),ANGH(100),SPRD(100)
COMMON /HILLS/ ECC(100),PXX(100),PY(100),PXY(100),BASE
COMMON /HILLS/ NHILLS
COMMON /COVER/ CX(150),CY(150),CPEAK(150),CPXX(150),CPYY(150)
COMMON /COVER/ CPXY(150),NCVELS
COMMON /COUNTR/KH,KHW,KV,KN,KGRS,KELL,KINT
COMMON /GRID/ LST(5,4),NHL(5,4),LISTH(150),KHREP(150),KTREP
COMMON /GRID/ LSTC(5,4),NC(5,4),LISTC(400),KCREP(150)
PAI=3.14159
L=5
READ(L,7) NHILLS
READ(L,47) BASE
FORMAT(10.4)
FORMAT(16)
FORMAT(6F10.3)
DO 50 I=1,NHILLS
  READ(L,17) XC(I),YC(I),PEAK(I),ANGH(I),SPRD(I),ECC(I)
CONTINUE
READ(L,37) LST
READ(L,37) NHL
READ(L,7) NHTOT
READ(L,37) (LISTH(I),I=1,NHTOT)
FORMAT(10I5)
DO 100 I=1,NHILLS
  AANGLE=ANGH(I)*PAI/180.
  SANG=SIN(AANGLE)
  CANG=COS(AANGLE)
  A=PEAK(I)/(PEAK(I)-50.)
  A=ALOG(A)
  B=A*ECC(I)**2
  SSPD=SPRD(I)**2
  PXX(I)=-(A*CANG*CANG+B*SANG*SANG)/SSPD
  PYY(I)=-(A*SANG*SANG+B*CANG*CANG)/SSPD
  PXY(I)=(2.*SANG*CANG*(B-A))/SSPD
  KHREP(I)=-2147483600
  
```

C  
 C  
 C  
 C  
 C  
 C

47  
 7  
 17  
 50  
 37

65



GRA13380  
GRA13390  
GRA13400  
GRA13410  
GRA13420  
GRA13430  
GRA13440  
GRA13450  
GRA13460  
GRA13470  
GRA13480  
GRA13490  
GRA13500  
GRA13510  
GRA13520  
GRA13530  
GRA13540  
GRA13550  
GRA13560  
GRA13570  
GRA13580  
GRA13590  
GRA13600  
GRA13610  
GRA13620  
GRA13630  
GRA13640  
GRA13650  
GRA13660  
GRA13670  
GRA13680  
GRA13690  
GRA13700  
GRA13710  
GRA13720  
GRA13730  
GRA13740  
GRA13750  
GRA13760  
GRA13770  
GRA13780  
GRA13790  
GRA13800  
GRA13810  
GRA13820  
GRA13830  
GRA13840  
GRA13850

```

C  ALL VALUES NOW IN METERS ON 0 -- 10,000 GRID
100 CONTINUE
  READ(L,7) NCVELS
  IF(NCVELS.EQ.0) GO TO 75
  DO 60 I=1,NCVELS
    READ(L,27) CXC(I),CYC(I),CPEAK(I),CPXX(I),CPYY(I),CPXY(I)
    FORMAT(3F10.4,3E13.7)
    KCREP(I)=-2147483600
  CONTINUE
  READ(L,37) LSTC
  READ(L,37) INC
  READ(L,7) NCTOT
  READ(L,37) (LISTC(I),I=1,NCTOT)
75  KTRIP=-2147483600
    KH=0
    KHW=0
    KV=0
    KN=0
    KGRS=0
    KELL=0
    KINT=0
    RETURN
  END

C  SUBROUTINE ROUTE
C
C  SUBROUTINE ROUTE COMPUTES THE ROUTE OF EACH ATTACKING UNIT
C  WHEN THE USER HAS SELECTED THE OPTION OF INPUTTING ATTACKER
C  ROUTES. IT CALCULATES THE COORDINATES OF EACH INTERVAL ENDPOINT
C  ALONG THE ROUTE, MAKING EACH INTERVAL LENGTH(DISTANCE MOVED DURING
C  A 10 SECOND TIME STEP) THE SAME. THE INTERVAL LENGTH IS DETERMINED
C  BY THE SPEED THE USER HAS SELECTED AND INPUTED FOR THE CURRENT
C  BATTLE.
C
COMMON /GRP3/ NBU,NRU,FL(6),F0(6),NOI(3),XIC(3,200),YIC(3,200),
1  IDIR(3,200),AVSP,ISPD
1  IUSTAT(6),I(6),LOST(6,6),VISFRA,VISFRB,SIZEK,
1  ISIZETW,NT(6),NF(6),SRF,DISMAX,
1  INLOSC(6,6),VISFR(6,6),RMINTK,RMXTK,RMINTW,RMXTW,OP,TOWER,TNKFR,
1  IPTT(3,3),RF,POA(6,6),APOA(6,6),LOA(6,6),OFL(6),POL(6)
1  DIMENSION XLOC(3,20),YLOC(3,20),N(3)
  IF(ISPD.EQ.4) DST=80.463
  IF(ISPD.EQ.3) DST=67.053
  IF(ISPD.EQ.2) DST=53.643
  IF(ISPD.EQ.1) DST=40.232
  LN=9
  DO 300 I=1,NRU
    READ(LN,15) N(I)

```



GRAL3860  
 GRAL3870  
 GRAL3880  
 GRAL3890  
 GRAL3900  
 GRAL3910  
 GRAL3920  
 GRAL3930  
 GRAL3940  
 GRAL3950  
 GRAL3960  
 GRAL3970  
 GRAL3980  
 GRAL3990  
 GRAL4000  
 GRAL4010  
 GRAL4020  
 GRAL4030  
 GRAL4040  
 GRAL4050  
 GRAL4060  
 GRAL4070  
 GRAL4080  
 GRAL4090  
 GRAL4100  
 GRAL4110  
 GRAL4120  
 GRAL4130  
 GRAL4140  
 GRAL4150  
 GRAL4160  
 GRAL4170  
 GRAL4180  
 GRAL4190  
 GRAL4200  
 GRAL4210  
 GRAL4220  
 GRAL4230  
 GRAL4240  
 GRAL4250  
 GRAL4260  
 GRAL4270  
 GRAL4280  
 GRAL4290  
 GRAL4300  
 GRAL4310  
 GRAL4320  
 GRAL4330

```

15  FORMAT(I2)
    NL=N(I)+1
    DO 200 IN=2,NL
      READ(LN,201) XLNCS,YLCS
201  FORMAT(F6.1,1X,F6.1)
      XLOC(I,IN)=XLNCS
      YLOC(I,IN)=YLCS
200  CONTINUE
      XLOC(I,1)=XIC(I,1)
      YLOC(I,1)=YIC(I,1)
      IDIR(I,1)=0
      NL=N(I)
      NUM=2
      DC 305 J=1,NL
      XL=XLOC(I,J+1)-XLOC(I,J)
      YL=YLOC(I,J+1)-YLOC(I,J)
      DIST=SQRT{XL**2+YL**2}
      Y=ABS(YL)
      Z=Y/XL
      ANGL=ATAN(Z)
      DEG=ANGL*57.2958
      IF(J.EQ.1) GO TO 320
      XLN=(DIST-EXTRA)*COS(ANGL)
      YLN=(DIST-EXTRA)*SIN(ANGL)
      XIC(I,NUM)=XIC(I,NUM-1)+XLN+XLE
      IF(YL.GT.0.) GO TO 325
      YLN=-YLN
325  YIC(I,NUM)=YIC(I,NUM-1)+YLN+YLE
      IF(YL.GT.0.) GO TO 340
      IDIR(I,NUM)=-IFIX(DEG)
      GO TO 341
      IDIR(I,NUM)=IFIX(DEG)
340  NUM=NUM+1
341  XLN=DST*COS(ANGL)
320  YLN=DST*SIN(ANGL)
      IF(YL.GT.0.) GO TO 310
      YLN=-YLN
310  IF(DIST.LT.DST) GO TO 315
      XIC(I,NUM)=XIC(I,NUM-1)+XLN
      YIC(I,NUM)=YIC(I,NUM-1)+YLN
      IF(YL.GT.0.) GO TO 342
      IDIR(I,NUM)=-IFIX(DEG)
      GO TO 343
      IDIR(I,NUM)=IFIX(DEG)
342  DIST=DIST-DST
343  NUM=NUM+1
      GO TO 310
  
```



```

315 EXTRA=DIST
XLE=EXTRA*COS(ANGL)
YLE=EXTRA*SIN(ANGL)
IF(YL.GT.0.) GO TO 305
YLE=-YLE
305 CONTINUE
300 CONTINUE
RETURN
END
C
C SUBROUTINE LAMDA(I,J,PCIVIS,DETRAT,PK)
C
C SUBROUTINE LAMDA IN CONJUNCTION WITH THE LOS ROUTINE COMPUTES
C THE DETECTION RATE(DETRAT) OF TARGET J BY THE OBSERVER I GIVEN
C THE PERCENT OF TARGET VISIBLE (PCIVIS) TO THE OBSERVER.
C
C COMMON /GRP1/ IPRDIR(6), ISECWD(6), MVTDIR(6), X(6), Y(6), SPD(6)
TCFACT=1.0
ZEROL=0.00001
PAI=3.14159
D=((ISECWD(1))*PAI/180.0)/2.0
BBB=((1.0/(2.0*(SIN(D))-D*COS(D))))
IF(ABS(BBB).LT.ZEROL) BBB=0.0
AAA=(-BBB)*COS(D)
IF(ABS(AAA).LT.ZEROL) AAA=0.0
OTANG=ATAN2((Y(J)-Y(I)),(X(J)-X(I)))
IF(OTANG.LT.-PAI/2.AND.OTANG.GT.-PAI) OTANG=2*PAI+OTANG
PD=IPRDIR(1)*PAI/180.0
IF((PD*OTANG).GE.0.0) GOTO 1
IF(PD.LT.0.0) GOTO 9
ANGLE=2*PAI+OTANG-PD
GOTO 10
9 ANGLE=2*PAI+PD-OTANG
10 IF(ANGLE.GT.PAI) ANGLE=2*PAI-ANGLE
1 GOTO 2
2 IF(ANGLE.GT.0) GOTO 3
DUP=PD+D
DLOW=PD-D
ANGLFT=OTANG+(15.0*PAI/180.)
IF(ANGLFT.GT.DUP) ANGLFT=DUP
ANGLRT=OTANG-(15.0*PAI/180.)
IF(ANGLRT.LT.DLOW) ANGLRT=DLOW
PK=BBB*ABS(ABS(SIN(ANGLFT))-ABS(SIN(ANGLRT)))+AAA*(ANGLFT-
1 ANGLRT)
IF(PK.LT.0.0) GOTO 3
IF(PK.GT.1.0) GOTO 5
GOTO 8

```

GRA14340  
 GRA14350  
 GRA14360  
 GRA14370  
 GRA14380  
 GRA14390  
 GRA14400  
 GRA14410  
 GRA14420  
 GRA14430  
 GRA14440  
 GRA14450  
 GRA14460  
 GRA14470  
 GRA14480  
 GRA14490  
 GRA14500  
 GRA14510  
 GRA14520  
 GRA14530  
 GRA14540  
 GRA14550  
 GRA14560  
 GRA14570  
 GRA14580  
 GRA14590  
 GRA14600  
 GRA14610  
 GRA14620  
 GRA14630  
 GRA14640  
 GRA14650  
 GRA14660  
 GRA14670  
 GRA14680  
 GRA14690  
 GRA14700  
 GRA14710  
 GRA14720  
 GRA14730  
 GRA14740  
 GRA14750  
 GRA14760  
 GRA14770  
 GRA14780  
 GRA14790  
 GRA14800  
 GRA14810





GRA14820  
 GRA14830  
 GRA14840  
 GRA14850  
 GRA14860  
 GRA14870  
 GRA14880  
 GRA14890  
 GRA14900  
 GRA14910  
 GRA14920  
 GRA14930  
 GRA14940  
 GRA14950  
 GRA14960  
 GRA14970  
 GRA14980  
 GRA14990  
 GRA15000  
 GRA15010  
 GRA15020  
 GRA15030  
 GRA15040  
 GRA15050  
 GRA15060  
 GRA15070  
 GRA15080  
 GRA15090  
 GRA15100  
 GRA15110  
 GRA15120  
 GRA15130  
 GRA15140  
 GRA15150  
 GRA15160  
 GRA15170  
 GRA15180  
 GRA15190  
 GRA15200  
 GRA15210  
 GRA15220  
 GRA15230  
 GRA15240  
 GRA15250  
 GRA15260  
 GRA15270  
 GRA15280  
 GRA15290

```

3  PK=0.0
   DETRAT=0.0
   GOTO 6
5  GK=1.0
8  RANGE=SQRT((X(J)-X(I))**2+(Y(J)-Y(I))**2)
   TOANG=ATAN2((Y(I)-Y(J)),(X(I)-X(J)))
   AD=MVTDIR(J)*PAI/180.0
   HORVEL=ABS(SPD(J)*SIN(TOANG-AD))
   HCRVEL=HORVEL*1609.3/3600.0
   DENOM=1.453+TCFACT*(0.5978+2.188*(RR**2)-0.5038*HORVEL)
   IF(DENOM.LE.ZEROL) DENOM=ZEROL
   DETRAT=0.003+1.088/DENOM
   DETRAT=DETRAT*PK
6  RETURN
   END

C  SUBROUTINE ELEV(X,Y,TMAC)
C
C  SUBROUTINE LEV DETERMINES THE TERRAIN ELEVATION FOR A GIVEN
C  SET OF X, Y COORDINATES. THIS FUNCTION IS USED IN CONJUNCTION
C  WITH THE LOS SUBROUTINE IN COMPUTING LINE-OF-SIGHT BETWEEN
C  OBSERVER AND TARGET.
C
COMMON /HILLS/ XC(100),YC(100),PEAK(100),ANGH(100),SPRD(100)
COMMON /HILLS/ ECC(100),PXX(100),PY(100),PXY(100),BASE
COMMON /HILLS/ NHILLS
COMMON /GRID/ LST(5,4),NHL(5,4),LISTH(150),KHREP(150),KTREP
COMMON /GRID/ LSTC(5,4),NC(5,4),LSTC(400),KCREP(150)
DATA GSIZE/1000./
C  FUNCTION TO COMPUTE TERRAIN ELEVATION FOR GIVEN X, Y COORDINATES.
ZMAX=BASE
IX=1+IFIX(X/GSIZE)
IY=1+IFIX(Y/GSIZE)
IF(NHL(IX,IY).EQ.0) GO TO 150
LS=LST(IX,IY)
LEND=LS+NHL(IX,IY)-1
DO 100 L=LS,LEND
  I=LISTH(L)
  QX=X-XC(I)
  QY=Y-YC(I)
  QSQ=QX*QX
  QYSQ=QY*QY
  QXY=QX*QY
  FACTOR=PX(X(I)*QXSQ+PY(Y(I)*QYSQ+PXY(I)*QXY
  IF(FACTOR.LT.-3.) GO TO 100
  HT=PEAK(I)*EXP(FACTOR)
  IF(HT.LE.ZMAX) GO TO 100

```



```

ZMAX=HT
CONTINUE
100 TMAC=ZMAX
150 RETURN
END

C
SUBROUTINE STOCH(I,RANGE,A)
C
C SUBROUTINE STOCH DETERMINES THE ATTRITION COEFFICIENTS WHEN
C A USER HAS SELECTED A STOCHASTIC ATTRITION OPTION. THE CALCULATION
C IS A FUNCTION OF THE ORIGINAL STOCHASTICALLY DETERMINED ATTRITION
C COEFFICIENT AS WELL AS A FUNCTION OF RANGE.
C
COMMON /GRP6/ ALPHA(6)
COMMON /GRP3/ NBU,FL(6),FO(6),NOI(3),XIC(3,200),YIC(3,200),
1 IDIR(3,200),AVSP,ISPD
1 IUSTAT(6),I(6),LOST(6,6),VISFRA,VISFRB,SIZEZK,
1 ISIZETW,NT(6),NF(6),SRF,DISMAX,
1 INLOSC(6,6),VISFR(6,6),RMINTK,RMXTK,RMINTW,RMXTW,OP,TOWER,TNKFR,
1 PTT(3,3),RF,POA(6,6),APOA(6,6),LOA(6,6),NA(6),OFL(6),POL(6)
1 IF(I.EQ.2) GO TO 5003
A=ALPHA(I)*((1.0-RANGE/RMXTW)**2)
GO TO 5004
5003 A=ALPHA(I)*((1.0-RANGE/RMXTK)**2)
5004 RETURN
END

C
SUBROUTINE ETK(I,RANGE,T)
C
C SUBROUTINE ETK COMPUTES THE EXPECTED TIME FOR A GIVEN FIRER TO
C KILL A GIVEN TARGET. THE CALCULATION IS A FUNCTION OF RANGE,
C TIME OF FLIGHT FOR A ROUND AND HIT AND KILL PROBABILITIES FOR
C THE FIRING WEAPON SYSTEM. IT IS A NUMBER THAT IS USED IN THE
C COMPUTATION OF THE DETERMINISTIC ATTRITION COEFFICIENTS.
C
COMMON /GRP2/ TA(2),T1(2),TH(2),TM(2),TF1(2),TF2(2),TF3(2),
1 P(2,6),PHH(2,6),PHM(2,6),PKH(2,6),TF(2)
1 IF(I.EQ.2) GOTO 5
TF(I)=TF1(I)
GOTO 6
5 IF(RANGE.GT.1000.0) GOTO 7
TF(I)=TF1(I)-(TF1(I)*(1000.0-RANGE)/1000.0)
GOTO 6
7 IF(RANGE.GT.2000.0) GOTO 8
TF(I)=TF2(I)-((TF2(I)-TF1(I))*(2000.0-RANGE)/1000.0)
GOTO 6
8 TF(I)=TF3(I)-((TF3(I)-TF2(I))*(3000.0-RANGE)/1000.0)
6 J=(RANGE+250.0)/500.0

```

GRA15300  
 GRA15310  
 GRA15320  
 GRA15330  
 GRA15340  
 GRA15350  
 GRA15360  
 GRA15370  
 GRA15380  
 GRA15390  
 GRA15400  
 GRA15410  
 GRA15420  
 GRA15430  
 GRA15440  
 GRA15450  
 GRA15460  
 GRA15470  
 GRA15480  
 GRA15490  
 GRA15500  
 GRA15510  
 GRA15520  
 GRA15530  
 GRA15540  
 GRA15550  
 GRA15560  
 GRA15570  
 GRA15580  
 GRA15590  
 GRA15600  
 GRA15610  
 GRA15620  
 GRA15630  
 GRA15640  
 GRA15650  
 GRA15660  
 GRA15670  
 GRA15680  
 GRA15690  
 GRA15700  
 GRA15710  
 GRA15720  
 GRA15730  
 GRA15740  
 GRA15750  
 GRA15760  
 GRA15770



GRA15780  
 GRA15790  
 GRA15800  
 GRA15810  
 GRA15820  
 GRA15830  
 GRA15840  
 GRA15850  
 GRA15860  
 GRA15870  
 GRA15880  
 GRA15890  
 GRA15900  
 GRA15910  
 GRA15920  
 GRA15930  
 GRA15940  
 GRA15950  
 GRA15960  
 GRA15970  
 GRA15980  
 GRA15990  
 GRA16000  
 GRA16010  
 GRA16020  
 GRA16030  
 GRA16040  
 GRA16050  
 GRA16060  
 GRA16070  
 GRA16080  
 GRA16090  
 GRA16100  
 GRA16110  
 GRA16120  
 GRA16130  
 GRA16140  
 GRA16150  
 GRA16160  
 GRA16170  
 GRA16180  
 GRA16190  
 GRA16200  
 GRA16210  
 GRA16220  
 GRA16230  
 GRA16240  
 GRA16250

```

IF(J.GT.6) J=6
T=TA(I)+TI(I)-TH(I)+(TH(I)+IF(I))/PKH(I,J)+(TM(I)+TF(I))/
IPHM(I,J))*((1.0-PHH(I,J))/PKH(I,J)+PHH(I,J)-P(I,J))
RETURN
END
C
SUBROUTINE SORT(I,M)
C
C
SUBROUTINE SORT IS USED TO SORT TARGETS IN ASCENDING RANGE
C ORDER. THIS IS USED TO DETERMINE THE PRIORITY OF A TARGET
C FOR FIRE ALLOCATION.
C
COMMON /GRP5/ LOT(6,6),ROT(6,6)
DO 19 J=1,M
IF(ROT(I,M).GE.ROT(I,J)) GOTO 19
21 R=ROT(I,J)
NN=LOT(I,J)
RCT(I,J)=ROT(I,M)
LOT(I,J)=LOT(I,M)
RCT(I,M)=R
LOT(I,M)=NN
19 CONTINUE
RETURN
END
C
SUBROUTINE KOVER(ZO,IMACT,SIZET,ZI,S,HTS,ZS,VISFRT)
C
C
SUBROUTINE KOVER DETERMINES WHAT PORTION OF A PARTICULAR TARGET
C IS COVERED BY THE TERRAIN BETWEEN THE TARGET AND OBSERVER.
C THIS NUMBER IS USED IN THE DETECTION AND ATTRITION COMPUTATION.
C
IF(S.EQ.0.) GO TO 2000
IF(HTS.GE.ZS) GO TO 2050
HEXT=ZO+(HTS-ZO)/S
EVI=AMAX1(HEXT,IMACT)
IF(EVIST.GE.ZI) GO TO 2050
IF(EVIST.LE.ZI-SIZET) RETURN
VIS=(ZI-EVIST)/SIZET
IF(VIS.LT.VISFRT) VISFRT=VIS
RETURN
2000 IF(HTS.LT.ZO) RETURN
2050 VISFRT=0.0
RETURN
END
C
SUBROUTINE LOS(XA,YA,TMACA,TMICA,SIZEA,XB,YB,TMACB,TMICB,SIZEB,
-LATOB,LBTOA,VISFRA,VISFRB)
C

```



```

C SUBROUTINE LOS COMPUTES A PERCENT OF A TARGET VISIBLE TO A
C PARTICULAR OBSERVER, GIVEN THE COORDINATES OF BOTH.
C
COMMON /HILLS/ XC(100),YC(100),PEAK(100),ANGH(100),SPRD(100)
COMMON /HILLS/ ECC(100),PXX(100),PY(100),PXY(100),BASE
COMMON /HILLS/ NHILLS
COMMON /COVER/ CXC(150),CYC(150),CPEAK(150),CPXX(150),CPYY(150)
COMMON /COVER/ CPXY(150),NCVELS
COMMON /COUNTR/KH, KHW, KV, KN KGRS, KELL, KINT
COMMON /GRID/ LST(5,4),NHL(5,4),LISTH(150),KHREP(150),KTREP
COMMON /GRID/ LSTC(5,4),NC(5,4),LISTC(400),KCREP(150)
COMMON /GRID/ LSTC(5,4),NC(5,4),LISTC(400),KCREP(150)
DIMENSION IGX(100),IGY(100),IEL(100),CS1(100),CS2(100)
DATA GSIZE/1000./
C SUBROUTINE TO COMPUTE FRACTION VISIBLE FOR OBSERVER TARGET PAIRS
VISFRA=1.
VISFRB=1.
XBA=XB-XA
YBA=YB-YA
IF((XBA.EQ.0.).AND.(YBA.EQ.0.)) RETURN
IF(SIZEA+TMICB.LE.0.) GO TO 510
IF(SIZEB+TMICB.LE.0.) GO TO 510
IF(TMICA.LT.0.) VISFRA=1.0+TMICA/SIZEA
IF(TMICB.LT.0.) VISFRB=1.0+TMICB/SIZEB
ZA=TMACA + TMICA + SIZEA
ZB=TMACB + TMICB + SIZEB
KTREP=KTREP+1
ZBA=ZB-ZA
XBASQ=XBA*XBA
YBASQ=YBA*YBA
XYBA=XBA*YBA
TWOXBA=2.*XBA
TWOYBA=2.*YBA
NGRSQ=0
IF(XBA) 110,95,100
95 XBA=0.1
100 ISGX=-1
XINC=GSIZE/XBA
GO TO 120
110 ISGX=1
XINC=-GSIZE/XBA
120 IF(YBA) 140,125,130
125 YBA=0.1
130 ISGY=-1
YINC=GSIZE/YBA
GO TO 150
140 ISGY=1
YINC=-GSIZE/YBA

```

GRA16260  
 GRA16270  
 GRA16280  
 GRA16290  
 GRA16300  
 GRA16310  
 GRA16320  
 GRA16330  
 GRA16340  
 GRA16350  
 GRA16360  
 GRA16370  
 GRA16380  
 GRA16390  
 GRA16400  
 GRA16410  
 GRA16420  
 GRA16430  
 GRA16440  
 GRA16450  
 GRA16460  
 GRA16470  
 GRA16480  
 GRA16490  
 GRA16500  
 GRA16510  
 GRA16520  
 GRA16530  
 GRA16540  
 GRA16550  
 GRA16560  
 GRA16570  
 GRA16580  
 GRA16590  
 GRA16600  
 GRA16610  
 GRA16620  
 GRA16630  
 GRA16640  
 GRA16650  
 GRA16660  
 GRA16670  
 GRA16680  
 GRA16690  
 GRA16700  
 GRA16710  
 GRA16720  
 GRA16730





GRA16740  
 GRA16750  
 GRA16760  
 GRA16770  
 GRA16780  
 GRA16790  
 GRA16800  
 GRA16810  
 GRA16820  
 GRA16830  
 GRA16840  
 GRA16850  
 GRA16860  
 GRA16870  
 GRA16880  
 GRA16890  
 GRA16900  
 GRA16910  
 GRA16920  
 GRA16930  
 GRA16940  
 GRA16950  
 GRA16960  
 GRA16970  
 GRA16980  
 GRA16990  
 GRA17000  
 GRA17010  
 GRA17020  
 GRA17030  
 GRA17040  
 GRA17050  
 GRA17060  
 GRA17070  
 GRA17080  
 GRA17090  
 GRA17100  
 GRA17110  
 GRA17120  
 GRA17130  
 GRA17140  
 GRA17150  
 GRA17160  
 GRA17170  
 GRA17180  
 GRA17190  
 GRA17200  
 GRA17210

```

150  IX=1+IFIX(XB/GSIZE)
      IY=1+IFIX(YB/GSIZE)
      XNEXT=GSIZE*(FLOAT(IX)+0.5*(ISGX-1.))
      YNEXT=GSIZE*(FLOAT(IY)+0.5*(ISGY-1.))
      XSTEP=(XB-XNEXT)/XBA
      YSTEP=(YB-YNEXT)/YBA
      NGRSQ=NGRSQ+1
      IGY(NGRSQ)=IX
      IF((XSTEP-GT.1.)-AND.(YSTEP-GT.1.)) GO TO 200
      IF(XSTEP-YSTEP) 170,180,190
170  IX=IX+ISGX
      XSTEP=XSTEP+XINC
      GO TO 160
180  IX=IX+ISGX
      XSTEP=XSTEP+XINC
190  IY=IY+ISGY
      YSTEP=YSTEP+YINC
      GO TO 160
200  KGRS=KGRS+NGRSQ
      C GRID SQUARE LIST NOW COMPLETE IN IGY, IGY WITH NGRSQ ENTRIES
      C
      C NOW FIND WHICH COVER ELLIPSES TOUCH THE A TO B LINE.
      C CHECK ELEVATIONS AT S1 AND S2 FOR EACH SUCH ELLIPSE.
      NELS=0
      CHTMAX=0.
      IF(NCVELS.EQ.0) GOTO 270
      DO 260 K=1,NGRSQ
        IX=IGX(K)
        IY=IGY(K)
        N=NC(IX,IY)
        IF(N.EQ.0) GO TO 260
        LS=LS+1
        LEND=LS+N-1
        DO 250 L=LS,LEND
          KELL=KELL+1
          IC=LISTC(L)
          IF(KCREP(IC).EQ.KTREP)
            KCREP(IC)=KTREP
          RX=XA-CXC(IC)
          RY=YA-CYC(IC)
          PPXX=CPXX(IC)
          PPYY=CPYY(IC)
          PPXY=CPXY(IC)
          AA=PPXX*XBASQ+PPYY*YBASQ+PPXY*XYBA
          BB=PPXX*XTWOXBA*RX+PPYY*XTWOYBA*RY+PPXY*(RX*YBA+RY*XBA)
          CC=PPXX*RX*RX+PPYY*RY*RY+PPXY*RX*RY-1.0
          ARG=BB*BB-4.0*AA*CC
  
```



GRA17220  
GRA17230  
GRA17240  
GRA17250  
GRA17260  
GRA17270  
GRA17280  
GRA17290  
GRA17300  
GRA17310  
GRA17320  
GRA17330  
GRA17340  
GRA17350  
GRA17360  
GRA17370  
GRA17380  
GRA17390  
GRA17400  
GRA17410  
GRA17420  
GRA17430  
GRA17440  
GRA17450  
GRA17460  
GRA17470  
GRA17480  
GRA17490  
GRA17500  
GRA17510  
GRA17520  
GRA17530  
GRA17540  
GRA17550  
GRA17560  
GRA17570  
GRA17580  
GRA17590  
GRA17600  
GRA17610  
GRA17620  
GRA17630  
GRA17640  
GRA17650  
GRA17660  
GRA17670  
GRA17680  
GRA17690

```

IF(ARG.LE.0.) GO TO 250
SQ=SQRT(ARG)
S1=-(BB+SQ)/(2.0*AA)
S2=(SQ-BB)/(2.0*AA)
IF(S1.GE.1.) GO TO 250
IF(S2.LE.0.) GO TO 250
IF(S1.LE.0.) GO TO 510
IF(S2.GE.1.) GO TO 510
CHECK LOS AT S1 AND S2
C NOW
KINT=KINT+1
CPK=CPEAK(IC)
XS=XAS+S2*XBA
YS=YAS+S2*YBA
CALL ELEV(XS,YS,HTS)
HTS=HTS+CPK
ZS=ZAS+S2*ZBA
IF(LATOB.EQ.0) GO TO 210
CALL KOVER(ZA,TMACB,SIZEB,ZB,S2,HTS,ZS,VISFRB)
IF(VISFRB.LE.0.) GO TO 510
IF(LBTQA.EQ.0) GO TO 220
S=1.-S2
CALL KOVER(ZB,TMACA,SIZEA,ZA,S,HTS,ZS,VISFRA)
IF(VISFRA.LE.0.) GO TO 510
XS=XAS+S1*XBA
YS=YAS+S1*YBA
CALL ELEV(XS,YS,HTS)
HTS=HTS+CPK
ZS=ZAS+S1*ZBA
IF(LATOB.EQ.0) GO TO 230
CALL KOVER(ZA,TMACB,SIZEB,ZB,S1,HTS,ZS,VISFRB)
IF(VISFRB.LE.0.) GO TO 510
IF(LBTQA.EQ.0) GO TO 240
S=1.0-S1
CALL KOVER(ZB,TMACA,SIZEA,ZA,S,HTS,ZS,VISFRA)
IF(VISFRA.LE.0.) GO TO 510
NELS=NELS+1
IEL(NELS)=IC
CS1(NELS)=S1
CS2(NELS)=S2
IF(CPK.GT.CHTMAX) CHTMAX=CPK
250 CONTINUE
260 CONTINUE
C ALL ELLIPSES CHECKED
C
C NOW START ON THE HILLS
270 DO 600 K=1,NGRSQ
IX=IGX(K)
IY=IGY(K)

```



GRA17700  
GRA17710  
GRA17720  
GRA17730  
GRA17740  
GRA17750  
GRA17760  
GRA17770  
GRA17780  
GRA17790  
GRA17800  
GRA17810  
GRA17820  
GRA17830  
GRA17840  
GRA17850  
GRA17860  
GRA17870  
GRA17880  
GRA17890  
GRA17900  
GRA17910  
GRA17920  
GRA17930  
GRA17940  
GRA17950  
GRA17960  
GRA17970  
GRA17980  
GRA17990  
GRA18000  
GRA18010  
GRA18020  
GRA18030  
GRA18040  
GRA18050  
GRA18060  
GRA18070  
GRA18080  
GRA18090  
GRA18100  
GRA18110  
GRA18120  
GRA18130  
GRA18140  
GRA18150  
GRA18160  
GRA18170

```

IF(NHL(IX,IY).EQ.0) GO TO 600
LS=LST(IX,IY)
LEND=LS+NHL(IX,IY) -1
DO 500 L=LS,LEND
I=LSTH(L)
IF(KHREP(I).EQ.KTREP) GO TO 500
KHREP(I)=KTREP
C PROCESSING FOR HILL I STARTS HERE
KH=KH+1
C COMPUTE W = TOP OF THIS HILL ALONG O-T LINE
C
TRX=XA-XC(I)
TRY=YA-YC(I)
TPXX=PX(X(I))
TPYY=PY(Y(I))
TPXY=PX(Y(I))
FQ=TWOXBA*TPXX*TRX+TWOYBA*TPYY*TRY+TPXY*(TRX*YBA+TRY*XBA)
GQ=TPXX*XBASQ+TPYY*YBASQ+TPXY*XYBA
IF(GQ.EQ.0.) GO TO 500
W=-FQ/(2.*GQ)
IF(ABS(W).GT.5.) GO TO 500
FSQ=FQ*FQ
EQ=TPXX*TRX+TPYY*TRY+TPXY*TRX*TRY
C
POWER=EQ-FSQ/(4.*GQ)
IF(POWER.LT.-3.) GO TO 500
HHW=PEAK(I)*EXP(POWER)
KHW=KHW+1
IF(HHW.LE.BASE) GO TO 500
ZW=ZA+W*ZBA
IF((W.LT.0.)OR.(W.GT.1.)) GO TO 300
IF(HHW.GE.ZW) GO TO 510
CVHTW=0.
IF(NELS.EQ.0) GO TO 300
DO 280 M=1,NELS
IF((CS1(M).GE.W).OR.(CS2(M).LE.W)) GO TO 280
IC=IEL(M)
IF(CVHTW.LT.CPEAK(IC)) CVHTW=CPEAK(IC)
280 CONTINUE
IF((HHW+CVHTW).GE.ZW) GO TO 510
300 IF(HHW+CHTMAX.LT.AMIN1(ZA-SIZEA,ZB-SIZEB)) GO TO 500
C IF WE GET TO HERE THEN NEED TO FIND LOWEST SIGHT LINE OVER HILL
C NEWTON ITERATION A TO B GIVING VISFRB
IF(LATOB.EQ.0) GO TO 400
KV=KV+1
V=W
HHV=HHW
NCT=0

```



GRA18180  
GRA18190  
GRA18200  
GRA18210  
GRA18220  
GRA18230  
GRA18240  
GRA18250  
GRA18260  
GRA18270  
GRA18280  
GRA18290  
GRA18300  
GRA18310  
GRA18320  
GRA18330  
GRA18340  
GRA18350  
GRA18360  
GRA18370  
GRA18380  
GRA18390  
GRA18400  
GRA18410  
GRA18420  
GRA18430  
GRA18440  
GRA18450  
GRA18460  
GRA18470  
GRA18480  
GRA18490  
GRA18500  
GRA18510  
GRA18520  
GRA18530  
GRA18540  
GRA18550  
GRA18560  
GRA18570  
GRA18580  
GRA18590  
GRA18600  
GRA18610  
GRA18620  
GRA18630  
GRA18640  
GRA18650

```

330 FV=FQ*V
      TWOGV=2.*GQ*V
      FCNV=ZA+HHV*(TWOGV*V+V-1.)
      KN=KN+1
      FACTOR=(TWOGV*TWOGV+2.*(GQ+TWOGV*FQ)+FSQ)
      DFCNV=HHV*V*FACTOR
      IF (ABS(DFCNV) .LT. 1.E-10) GO TO 350
      V=V-FCNV/DFCNV
      FV=FQ*V
      TWOGV=2.*GQ*V
      POWER = EQ+V+GQ*V*V
      IF (POWER .LT. -3.) GO TO 400
      HHV=PEAK(I)*EXP(POWER)
      DHHV=HHV*(FQ+TWOGV)
      ELV=ZA+DHHV*V
      IF (ABS(HHV-ELV) .LT. 1.) GO TO 350
      NCT=NCT+1
      IF (NCT .LT. 10) GO TO 330
      IF ((V .LT. 0.) .OR. (V .GT. 1.)) GO TO 400
      CVHTV=0.
      IF (NELS .EQ. 0) GO TO 390
      DO 380 M=1, NEL
      IF ((CS1(M).GE.V) .OR. (CS2(M).LE.V)) GO TO 380
      IC=IEL(M)
      IF (CVHTV .LT. CPEAK(IC)) CVHTV=CPEAK(IC)
      CONTINUE
      HTV=HHV+CVHTV
      ZV=ZA+V*ZBA
      CALL KOVER(ZA,TMACB,SIZEB,ZB,V,HTV,ZV,VISFRB)
      IF (VISFRB.LE.0.) GO TO 510
      C NEWTON ITERATION B TO A GIVING VISFRA
      400 IF (ABS(V).GT.5.) GO TO 400
      IF (LBTOA.EQ.0) GO TO 500
      KV=KV+1
      V=W
      VM1=V-1.
      HHV=HHW
      NCT=0
      FV=FQ*V
      TWOGV=2.*GQ*V
      FCNV=ZB+HHV*((FQ+TWOGV)*VM1-1.)
      KN=KN+1
      FACTOR=(TWOGV*TWOGV+2.*(GQ+TWOGV*FQ)+FSQ)
      DFCNV=HHV*VM1*FACTOR
      IF (ABS(DFCNV) .LT. 1.E-10) GO TO 450
      V=V-FCNV/DFCNV
      IF (ABS(V).GT.5.) GO TO 500
      VM1=V-1.

```





GRA18660  
GRA18670  
GRA18680  
GRA18690  
GRA18700  
GRA18710  
GRA18720  
GRA18730  
GRA18740  
GRA18750  
GRA18760  
GRA18770  
GRA18780  
GRA18790  
GRA18800  
GRA18810  
GRA18820  
GRA18830  
GRA18840  
GRA18850  
GRA18860  
GRA18870  
GRA18880  
GRA18890  
GRA18900  
GRA18910  
GRA18920  
GRA18930  
GRA18940  
GRA18950

```

FV=FQ*V
TWOGV=2.*GQ*V
POWER = EQ+FV+GQ*V*V
IF (POWER .LT. -3.) GO TO 500
HHV=PEAK(I)*EXP(POWER)
DHHV=HHV*(FQ+TWOGV)
ELV=ZB+DHHV*VM1
IF (ABS(HHV-ELV) .LT. 1.) GO TO 450
NCT=NCT+1
IF (NCT.LT.10) GO TO 430
IF ((V.LT.0.).OR.(V.GT.1.)) GO TO 500
CVHTV=0.
IF (NELS.EQ.0) GO TO 490
DO 480 M=1,NELS
IF ((CS1(M).GE.V).OR.(CS2(M).LE.V))GO TO 480
IC=IEL(M)
IF (CVHTV.LT.CPEAK(IC)) CVHTV=CPEAK(IC)
CONTINUE
HTV=HHV+CVHTV
ZV=ZA+V*ZBA
S=-VM1
CALL KOVER(ZB,TMACA,SIZEA,ZA,S,HTV,ZV,VISFRA)
IF (VISFRA.LE.0.) GO TO 510
CONTINUE
CONTINUE
RETURN
VISFRA=0.
VISFRB=0.
RETURN
END

```

450

480

490

500

600

510



# PLOTTING PROGRAM FOR TERRAIN CONTOUR LINE

96



XXH(4)=5000.  
YYH(4)=1000.

CC C USING THE BISECTION SEARCH METHOD, FIND THE LOCATION  
C WHICH HAS A ELEVATION, I, E. 20, 40, ...ETC.

ME=5  
YELTA=20.  
XELTA=5.  
XEL=1000.  
YEL=1000.  
YCON=1000  
XEND=5000  
YEND=4000.  
991 CALL ELEV(XEL,YEL,ZNEW)  
TELTA=YELTA  
NVAL1=IFIX(ZNEW/20.)  
717 ZOLD=ZNEW  
727 YEL=YEL+YELTA  
IF(YEL.LT.YEND) GO TO 777  
XEL=XEL+5.  
IF(XEL.EQ.XEND) GO TO 333  
YEL=YCON  
GO TO 991  
777 CALL ELEV(XEL,YEL,ZNEW)  
NVAL2=IFIX(ZNEW/20.)  
IZZ=IABS(NVAL2-NVAL1)  
IF(IZZ.GE.1) GO TO 788  
NVAL1=NVAL2  
GO TO 717  
788 KVAL=NVAL2  
YRES=YEL  
IF(NVAL2.GT.NVAL1) GO TO 630  
ZVAL=NVAL1\*20.  
NK=1  
TELTA=YELTA  
631 YEL=YEL-TELTA/2.  
632 CALL ELEV(XEL,YEL,ZNEW)  
IF(ZNEW.LT.ZVAL+0.1.AND.ZNEW.GT.ZVAL-0.1) GO TO 800  
IF(NK.GT.4) GO TO 800  
ZOLD=ZNEW  
TELTA=TELTA/2.  
NK=NK+1  
IF(ZNEW.LT.ZVAL) GO TO 631  
YEL=YEL+TELTA/2.0  
GO TO 632

CC C CALCULATE INCREASING ELEVATION

630 ZVAL=NVAL2\*20.  
NK=1  
TELTA=YELTA  
642 YEL=YEL-TELTA/2.0  
643 CALL ELEV(XEL,YEL,ZNEW)  
IF(ZNEW.LT.ZVAL+0.1.AND.ZNEW.GT.ZVAL-0.1) GO TO 800  
IF(NK.GT.4) GO TO 800  
ZOLD=ZNEW  
TELTA=TELTA/2.0  
NK=NK+1  
IF(ZNEW.GT.ZVAL) GO TO 642  
YEL=YEL+TELTA/2.0  
GO TO 643

CC C COLLECT ELEVATION COORINATE DATA WHICH IS WANTED TO BE  
C PLOTTED.

800 IF(ZVAL.LT.20.) GO TO 189



```

IF(ZVAL.GT.140.) GO TO 189
IF(ZVAL.NE.20.AND.ZVAL.LT.100.) GO TO 189
XXH(ME)=XEL
YYH(ME)=YEL
ME=ME+1
189 NVAL1=KVAL
YEL=YRES
GO TO 727
333 NP=ME-1
CALL PLOTG(XXH,YYH,NP,1,0,75,'X-AXIS LABEL',12,
1 'Y-AXIS LABEL',12,XMIN,XMAX,YMIN,YMAX,8.,6.)
CALL PLOT(0.,0.,999)
STOP
END
SUBROUTINE ELEV(X,Y,TMAC)

```

C  
C  
C

COMPUTE THE ELEVATION FOR A GIVEN X,Y COORDINATE

```

IMPLICIT REAL*4(A-H,O-Z)
COMMON /HILLS/ XC(100),YC(100),PEAK(100),ANGH(100)
COMMON /HILLS/ SPRD(100),ECC(100),PXX(100),PYY(100)
COMMON /HILLS/ PXY(100),BASE,NHILLS
COMMON /GRID/ LST(5,4),NHL(5,4),LISTH(100),KHREP(100)
COMMON /GRID/ KTREP
DATA GSIZE/1000./
ZMAX=BASE
IX=1+IFIX(X/GSIZE)
IY=1+IFIX(Y/GSIZE)
IF(NHL(IX,IY).EQ.0) GO TO 150
LS=LST(IX,IY)
LEND=LS+NHL(IX,IY)-1
DO 100 L=LS,LEND
I=LISTH(L)
QX=X-XC(I)
QY=Y-YC(I)
QXSQ=QX*QX
QYSQ=QY*QY
QXY=QX*QY
FACTOR=PXX(I)*QXSQ+PYY(I)*QYSQ+PXY(I)*QXY
IF(FACTOR.LT.-3.) GO TO 100
HT=PEAK(I)*EXP(FACTOR)
IF(HT.LE.ZMAX) GO TO 100
ZMAX=HT
100 CONTINUE
150 TMAC=ZMAX
RETURN
END

```





## LIST OF REFERENCES

1. Bonder, S., "The Lanchester Attrition-Rate Coefficient," Operations Research, v. 15, p. 221-232, May 1967.
2. Bonder, S., "An Overview of Land Battle Modeling in the U.S." In: Proceedings 13th U.S. Army Operations Research Symposium, pp. 73-77, 1974.
3. Chardwick, D.L., The Evaluation of Design and Employment Alternative for the LVA: A Modelling Strategy, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1978.
4. Farmer, W.T., A Survey of Approaches to the Modeling of Land Combat, M.S. Thesis, Naval Postgraduate School, Monterey, California, June 1980.
5. Hartman, J.K., Parametric Terrain and Line-of-Sight Modeling in the STAR Combat Model, Naval Postgraduate School, Monterey, California, 1979.
6. Mills, G.M., The Enrichment of Smoller's Model of Land Combat, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1980.
7. Morris, W.T., "On the Art of Modeling," Management Science 13, B707-B717, August 1967.
8. Needels, C.G., Parameterization of Terrain in Army Combat Models, M.S. Thesis, Naval Postgraduate School, Monterey, California, March 1976.
9. Smoller, J., Operational Lanchester-type Model of Small Unit Land Combat, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1979.
10. Taylor, J.G., Force-on-Force Attrition Modeling, Military Application Section of Operations Research of America, January 1980.
11. Wallace, W.S., and Hagewood, E.G., Simulation of Tactical Alternative Responses (STAR), M.S. Thesis, Naval Postgraduate School, Monterey, California, December 1978.



INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3. Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4. Professor James G. Taylor, Code 55Tw Department of Operations Research Naval Postgraduate School Monterey, California 93940	12
5. Major Jeffrey L. Elles, Code 55EI Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
6. Operations Research Department Deputy Chief of Staff for Strategy and Plans ROKA HQ, Seoul, Korea	1
7. LTC Park Soon Dae Office of Operations Research/System Analysis ROKN HQ, Seoul, Korea	2



Thesis

194427

P15757 Park

c.1

An operational

lanchester-type model

of small-unit amphi-

bious operations.

29 SEP 85

1 OCT 84

27546

13449

Thesis

194427

P15757 Park

c.1

An operational

lanchester-type model

of small-unit amphi-

bious operations.

thesP15757

An operational lanchester-type model of



3 2768 001 00055 7  
DUDLEY KNOX LIBRARY